

UNIVERSIDAD CARLOS III DE MADRID

ESCUELA POLITÉCNICA SUPERIOR

INGENIERÍA DE TELECOMUNICACIÓN

Departamento de Teoría de la Señal y Comunicaciones



PROYECTO FINAL DE CARRERA

**IMPLEMENTACIÓN DE UN SISTEMA DE COMUNICACIÓN
EN UN DISPOSITIVO RADIO**

AUTOR: RONY FONSECA ARBOLEDA

TUTORA: ANA GARCÍA ARMADA

Leganés, Febrero de 2012

TÍTULO: *IMPLEMENTACIÓN DE UN SISTEMA DE COMUNICACIÓN EN UN DISPOSITIVO RADIO.*

AUTOR: *Rony Fonseca Arboleda*

TUTORA: *Ana García Armada*

La defensa del presente Proyecto Fin de Carrera se realizó el día 27 de Febrero de 2012; siendo calificada por el siguiente tribunal:

PRESIDENTE: *Matilde Sánchez Fernández*

SECRETARIO *Víctor Gil Jiménez*

VOCAL *Celia López Ongil*

Habiendo obtenido la siguiente calificación:

CALIFICACIÓN:

Presidente

Secretario

Vocal

Agradecimientos

A mi Madre por su ejemplo de lucha y superación, siempre ha estado en los momentos difíciles y ha hecho posible que esto ocurra. A mi familia que ha soportado mis constantes ausencias durante todo este tiempo.

A mis amigos y compañeros de la Universidad, a los primeros Paola, Dario, Dani, Roberto por haberme acogido como uno más y a los que compartieron conmigo los últimos años de carrera en especial a Ernesto, Andrés, Hugo y Marcos juntos compartimos muchas tristezas pero juntos hemos compartido muchísimas más alegrías, fiestas, viajes y cenas piratas.

A todas las personas que durante este largo periodo de aprendizaje han dejado huella en mi, a los profesores que han sabido transmitir sus conocimientos y que han contribuido a mi formación académica y personal.

Quiero agradecer especialmente a Ana García Armada por brindarme la oportunidad de realizar este proyecto.

Al personal de vigilancia del edificio del parque científico de la universidad Carlos III de Madrid que me aguantaron muchos fines de semana.

A Dios por su fidelidad y por permitir enrolarme en la caprichosa travesía de estudiar Ingeniería de Telecomunicación.

*El genio es un uno por ciento de inspiración
y un noventa y nueve por ciento de sudor.*

Thomas Alva Edison

Resumen

El presente proyecto fin de carrera de Ingeniería de Telecomunicación tiene como objetivo dar continuidad a un línea de trabajo que busca diseñar un sistema de comunicación radio, que utiliza una técnica de modulación de múltiples portadoras ortogonales (OFDM), sobre un dispositivo *Software Defined Radio*, con el fin de realizar estudios sobre las señales transmitidas.

La creciente demanda de movilidad y de comunicaciones que exigen gran ancho de banda ha forzado el desarrollo de tecnologías software y hardware que dan soporte a tal fin. Los dispositivos SDR diseñados para que todas o algunas de las funciones de la capa física sean definidas mediante software permiten una gran adaptabilidad en las comunicaciones radio. Dicha flexibilidad radica en la posibilidad de desarrollar estándares de comunicación inalámbrica sobre una FPGA sin necesidad de realizar modificaciones físicas en el dispositivo.

En esta fase del proyecto global se parte de un diseño inicial de un modulador OFDM descrito en lenguaje de alto nivel VHDL. Se busca pues mejorar el diseño original y estructurar cada una de las partes del sistema completo, diferenciando entre la etapa de transmisión y la de recepción. Para que esta segunda etapa se pueda realizar se diseñó un demodulador digital en cuadratura.

Este diseño se implementa sobre un dispositivo *Small Form Factor SDR* de la marca Lyrtech que cuenta con un módulo de procesamiento digital, un módulo de adquisición de datos y un módulo de radiofrecuencia.

Para la realización de este trabajo se han empleado las plataforma de desarrollo *ISE Foundation* y *Code Composer Studio* y para la validación y simulación de cada módulo desarrollado se ha utilizado *Matlab* y *Modelsim*. Se han llevado a cabo diversas medidas en tiempo y frecuencia sobre la señal transmitida y recibida que han permitido validar el diseño y verificar los límites de funcionamiento del dispositivo.

Índice general

Introducción	23
1. Conceptos Generales	27
1.1. Comunicación	27
1.2. Canal de comunicación	28
1.2.1. Atenuación	29
1.2.2. Desvanecimiento por multitrayecto	29
1.2.3. Desvanecimiento selectivo en frecuencia	30
1.3. Procesamiento digital de la señal	32
1.4. Modulación	32
1.4.1. ¿Por qué modulamos?	33
1.5. Filtro digital	34
1.5.1. Filtro IIR	35
1.5.2. Filtro FIR	35
1.5.3. IIR frente a FIR	36
2. Entorno de Trabajo	37
2.1. Herramientas Software	37
2.1.1. Code Composer Studio	37
2.1.2. Xilinx ISE Foundation	38
2.1.3. Otros programas	39
2.2. Herramientas Hardware	39

2.2.1. Módulo de procesamiento digital	40
2.2.2. Módulo de conversión de datos	42
2.2.3. Módulo de radiofrecuencia	42
3. Modelo de Prueba	45
3.1. ¿Que se va a implementar?	45
3.2. Modulación OFDM	46
3.3. Descripción del sistema	47
3.3.1. Estructura	48
3.3.2. Modelo de la señal	48
3.3.3. Formato de trama	49
3.4. Fase inicial	50
3.5. Fase actual	50
3.5.1. Nivel Software	52
3.5.2. Nivel Hardware	52
3.5.3. Nivel Diseño	53
4. Demodulador Digital en Cuadratura	57
4.1. Motivación	57
4.2. Prototipo	58
4.3. Validación del modelo	59
4.3.1. Simulación con Matlab	60
4.3.2. Utilizando IP Cores	61
4.4. Optimización de recursos	64
4.4.1. Análisis teórico	64
4.4.2. Diseño para FPGA	65
5. Pruebas y Resultados	69
5.1. Transmisión OFDM	69
5.1.1. Medidas en Banda Base	69
5.1.2. Señal radio	72
5.1.3. Señal recibida a FI	72
5.2. Funcionalidad de la herramienta	75

5.3. Simulación del demodulador IQ	81
5.3.1. Vamos por partes	81
5.3.2. Optimizando recursos	83
6. Conclusiones	85
6.1. Líneas futuras	86
6.2. Dificultades	86
APÉNDICES	89
A. Presupuesto Del Proyecto	91
B. Diseño de Módulo Hardware sobre FPGA	93
B.1. Creación del diseño	94
B.1.1. Componentes IP	95
B.1.2. Simulación funcional	95
B.2. Síntesis	96
B.3. Implementación	96

Lista de Figuras

1.1. Representación del modelo de Comunicación	28
1.2. Representación del multitrayecto	30
1.3. Clasificación de las formas de modulación	33
2.1. Flujo de desarrollo del código en c/c++	38
2.2. Flujo de desarrollo hardware	39
2.3. Plataforma SFF SDR.	40
2.4. Bloques del módulo DSP.	41
2.5. Bloques del módulo de conversión de datos.	43
2.6. Bloques del módulo de Radiofrecuencia.	44
3.1. Trama OFDM en tiempo y frecuencia.	47
3.2. Diagrama de Bloques Sistema OFDM.	48
3.3. Estado Inicial.	51
3.4. Bloque Transmisor.	54
3.5. Simulación del Transmisor.	55
4.1. Demodulador IQ convencional.	58
4.2. Estructura Demodulador IQ Digital.	59
4.3. Módulo y fase del filtro y de la componente I.	60
4.4. Configuración del NCO.	62
4.5. Configuración del multiplicador.	63
4.6. Esquemático de una instancia del filtro FIR.	63

4.7. Diagrama de bloques del demodulador IQ implementado.	66
4.8. Recuperación de la componente I.	67
4.9. Recuperación de la componente Q.	67
5.1. Frecuencia del reloj del transmisor.	70
5.2. Secuencia corta de entrenamiento.	71
5.3. Trama de 100 símbolos OFDM.	71
5.4. Tiempo de procesamiento del modulador.	72
5.5. Espectro de la señal Banda-Base.	73
5.6. Señal RF.	73
5.7. Espectro de la señal transmitida.	74
5.8. Espectro de la señal transmitida (Ampliación).	74
5.9. Señal a frecuencia intermedia.	75
5.10. Atenuación de la señal recibida.	76
5.11. Espectro de la señal recibida.	76
5.12. Ampliación del espectro de la señal recibida.	77
5.13. Frecuencia de transmisión mayor de 60 MHz.	78
5.14. Transmisión a 740 MHz.	78
5.15. Transmisión a 480 MHz.	79
5.16. Transmisión a 1,1 GHz.	80
5.17. Simulación del NCO.	82
5.18. Simulación del Multiplicador.	82
5.19. Simulación del Filtro.	83
5.20. Simulación del Demodulador Digital implementado.	84
B.1. Flujo de diseño para FPGAs de Xilinx en VHDL.	94
B.2. Flujo del proceso de implementación.	96

Lista de Tablas

4.1. Recursos utilizados por demodulador	63
4.2. Recursos utilizados por demodulador implementado	68
5.1. Amplificación DAC	80
A.1. Fases del Proyecto	91
A.2. Costes de material	92
A.3. Presupuesto	92

Lista de Acrónimos

ADC Analog to Digital Converter. 34, 40, 42

ADSL Asymmetric Digital Subscriber Line. 45

API Application Programming Interface. 41, 52

BER Bit Error Rate. 31

BSDK Board Software Development kit. 37

CCS Code Composer Studio. 37

CIR Carrier to Interference Ratio. 31

DAC Digital to Analog Converter. 34, 40, 42

DSP Digital Signal Processing. 37

DVB-T Digital Video Broadcasting-Terrestrial. 45

EDIF Electronic Design Interchange Format. 38, 96

EMIF External Memory Interface. 41

FFT Fast Fourier Transform. 46

FIR Finite Impulse Response. 11, 34, 35

FPGA Field Programmable Gate Array. 37, 40, 41

FRS Family Radio Service. [44](#), [77](#)

GMRS General Mobile Radio Service. [44](#), [77](#)

ICI Inter Carrier Interference. [47](#)

IEEE Institute of Electrical and Electronics Engineers. [49](#)

IFFT Inverse Fast Fourier Transform. [46](#)

IIR Infinite Impulse Response. [11](#), [34](#), [35](#)

LSB Lower Side Band. [44](#)

LTE Long Term Evolution. [45](#)

MSPS Mega Samples Per Second. [42](#)

NCD Native Circuit Design. [97](#)

NCF Native Constraints File. [38](#), [96](#)

NCO Numerically Controlled Oscillator. [42](#)

OFDM Orthogonal Frequency Division Multiplexing. [33](#), [45](#), [46](#), [48](#), [50](#)

QPSK Quadrature Phase-Shift Keying. [50](#)

RFID Radio Frequency Identification. [45](#)

SDR Software Defined Radio. [37](#), [40](#), [45](#)

SNR Signal to Noise Ratio. [31](#)

VCO Voltage-Controlled Oscillator. [43](#)

VHDL VHSIC Hardware Description Language. [37](#)

VITAL VHDL Initiative Towards ASIC Library. [95](#)

VLSI Very Large Scale Integration. [34](#)

VPSS Video Processing Subsystem. [41](#)

Wimax Worldwide Interoperability for Microwave Access. [45](#)

XNF Xilinx Netlist Format. [38](#), [96](#)

Introducción

El mundo está en constante evolución y de este desarrollo surge la necesidad de una comunicación sin ataduras que permita el libre movimiento de las personas. En la última década las comunicaciones inalámbricas han experimentado un auge en su demanda y en consecuencia en su estudio y explotación. El vertiginoso avance tecnológico ha permitido la creación de potentes aparatos que incluyen toda la electrónica necesaria para la transmisión de la información por cualquier canal.

El dispositivo SDR (Software Defined Radio) es una de tantas herramientas que han sido diseñadas para ofrecer comunicaciones en entornos radio, gracias a su potente módulo de procesamiento digital de señales, su módulo de conversión de datos y su módulo de radiofrecuencia, que permite la transmisión de señales a altas frecuencias, así como su tamaño compacto, hacen de este componente una unidad portátil ideal para este tipo de comunicaciones.

Pero la movilidad trae consigo la exigencia de comunicaciones de banda ancha y en la actualidad estándares como el IEEE 802.11a se basan en técnicas de acceso múltiple por división de frecuencias ortogonales para alcanzar una tasa de transmisión elevada.

Uno de los factores limitantes en sistemas de comunicaciones móviles inalámbricas es la interferencia intersimbólica (ISI) debida al multitrayecto. En sistemas de una única portadora la duración de símbolo es muy pequeña ocupando un gran ancho de banda y el efecto del multitrayecto en distintos instantes de tiempo recae sobre múltiples símbolos, produciendo la ISI. Es por esto que el sistema OFDM se presenta como alternativa factible para mitigar estos efectos y satisfacer los requerimientos de velocidad y eficiencia espectral.

Objetivos

El presente trabajo busca darle continuidad al desarrollo iniciado en el proyecto *IMPLEMENTACIÓN DE UN SISTEMA OFDM EN UN DISPOSITIVO SFF SDR* [15], mejorando el diseño original. Implementar la nueva versión en el dispositivo SDR configurando todos sus módulos y de esta forma conseguir un ciclo completo de transmisión y recepción de la información.

Mediante este estudio se pretende conocer las bondades de un sistema de transmisión multiportadora utilizando un dispositivo de comunicación radio, explorando todas las posibilidades que esta herramienta ofrece. Para llevar a cabo esta tarea se diseñará un demodulador en cuadratura orientado a consumir el mínimo de recursos disponibles y se realizarán diversas pruebas que validen su funcionamiento así como medidas en tiempo y frecuencia de las señales generadas.

Alcance

Con este proyecto se creará una metodología de trabajo por módulos para el diseño e implementación de un sistema de transmisión de múltiples portadoras ortogonales. Se buscará mayor eficiencia y mejor utilización de recursos ajustando el diseño original.

Se explorarán las diversas configuraciones que el dispositivo de transmisión permite, configurando cada uno de sus módulos. Se utilizarán las herramientas software *ISE Foundation* y *Code Composer Studio* para las mejoras en el diseño, el desarrollo de nuevos módulos y su posterior implementación. Se utilizará el programa *Modelsim* para simular todos los bloques del sistema.

Para completar el bloque receptor se diseñará un demodulador en cuadratura en lenguaje de alto nivel VHDL. Por último, se realizarán pruebas medibles mediante un osciloscopio de la señal transmitida y recibida.

Trabajos previos

Como se mencionó anteriormente, este trabajo surge por la necesidad de dar continuación al proyecto *IMPLEMENTACIÓN DE UN SISTEMA OFDM EN UN DISPOSITIVO SFF SDR* [15], en dicho estudio se diseñó, en lenguaje VHDL, un modulador que permite la transmisión de datos en tramas OFDM como lo describe la publicación *Design and Implementation of Synchronization and AGC for OFDM-based WLAN Receivers* [4] y se realizó una configuración básica de los módulos del dispositivo SDR. Obteniendo como resultado una señal transmitida a

la frecuencia por defecto del módulo de radiofrecuencia.

Siguiendo esta línea de trabajo se utilizará el diseño del modulador para generar una versión mejorada del sistema y completar la configuración de la herramienta explorando toda la banda de transmisión que permite el módulo de radiofrecuencia. Así como dotar al sistema de los elementos necesarios para concluir el ciclo de recepción.

Contexto

Este proyecto fin de carrera se ha realizado en el departamento de Teoría de la Señal y Comunicaciones de la Universidad Carlos III de Madrid y está enmarcado dentro de una línea de trabajo que busca poner en práctica algunas de las investigaciones realizadas por profesores del departamento y que hasta el momento sólo se habían podido probar mediante simulación.

En particular el trabajo desarrollado está enfocado en la implementación, sobre un dispositivo *Software Defined Radio*, del modelo de un sistema de transmisión de múltiples portadoras.

La mayor parte de este trabajo ha tenido lugar en el *Laboratorio de Sistemas de Comunicaciones para Seguridad y Espacio* situado en el *Parque Científico de la Universidad Carlos III de Madrid*.

Contenido de la memoria

A continuación se describe de forma resumida cada uno de los capítulos que forman parte de esta memoria, a saber:

Capítulo 1: En éste se desarrollan conceptos útiles para la comprensión de los temas tratados en el presente proyecto, de esta forma brindar una visión global de todos los aspectos que relacionan las comunicaciones que se realizan a través de un medio radio.

En la primera parte se trata el tema de la comunicación, a continuación se exponen los fenómenos que puede sufrir la transmisión de la información por un medio radio y finalmente se habla sobre algunos conceptos relacionados con el procesado digital de señales.

Capítulo 2: Describe el entorno de trabajo, empieza con la descripción de los programas utilizados para el diseño, simulación e implantación del sistema OFDM y continúa con la especificación de los componentes físicos más relevantes del dispositivo SFF SDR.

Capítulo 3: Detalla los motivos que impulsaron el desarrollo de un sistema de transmisión

multiportadora, presenta un resumen de la estructura utilizada para realizar las pruebas de transmisión y recepción de la información y por último describe las mejoras y los avances conseguidos. Además, pone en contexto el trabajo realizado aclarando cuál ha sido el punto de partida.

Capítulo 4: En él se presenta el proceso de diseño e implementación de un Demodulador Digital en Cuadratura sobre un dispositivo SDR. Discute la motivación de este desarrollo, plantea los escenarios de varias soluciones y finalmente describe el modelo realizado en VHDL.

Capítulo 5: En este capítulo se exponen los resultados obtenidos de las mediciones realizadas sobre el sistema desarrollado y las pruebas de funcionamiento de la herramienta física. Empieza mostrando las características de la señal en sus tres fases (banda base, radiofrecuencia y frecuencia intermedia), continúa examinando las prestaciones y limitaciones del dispositivo, y termina con la evaluación del demodulador digital.

Capítulo 6: Es el capítulo de las conclusiones, en él se hace una descripción resumida y una valoración crítica del conjunto del trabajo realizado. Se detallan las aportaciones de este proyecto al propósito global evaluando el cumplimiento de los objetivos. Se plantean posibles ampliaciones y mejoras sobre el trabajo desarrollado y finaliza con el relato de las incidencias mas relevantes ocurridas durante este trabajo.

Capítulo 1

Conceptos Generales

En este capítulo se presentan conceptos relacionados con la comunicación y la transmisión de información a través de un medio radio, así como nociones de algunas de las técnicas realizadas en el procesado digital de la señal.

1.1. Comunicación

La comunicación es una realidad emergente, un estado de cosas excepcionales. Surge a través de una síntesis de tres selecciones diferentes, a saber, la selección de la información, la selección de la emisión de esta información, y una comprensión o mala interpretación selectiva de esta emisión y la información que contiene [5]. Cada uno de estos componentes es esencial para lograr la comunicación. En otras palabras, cualquier forma de comunicación requiere de un emisor, un mensaje a transmitir y un destinatario de dicho mensaje ó receptor, pero este último no necesita estar presente ni tiene que ser consciente del acto comunicativo del emisor para que la comunicación se lleve a cabo. En este proceso, la información es enviada por el emisor a través de un canal hacia el receptor quien decodifica el mensaje y genera una respuesta.

En el ámbito de las telecomunicaciones el modelo básico de comunicación fue postulado en 1949 gracias a un trabajo conjunto entre Claude E. Shannon y Warren Weaver. Este modelo describe la comunicación como un proceso sistémico en el cual sus principales componentes son:

- Emisor: Es quien enuncia el mensaje en el acto de comunicación, puede ser o no una persona.

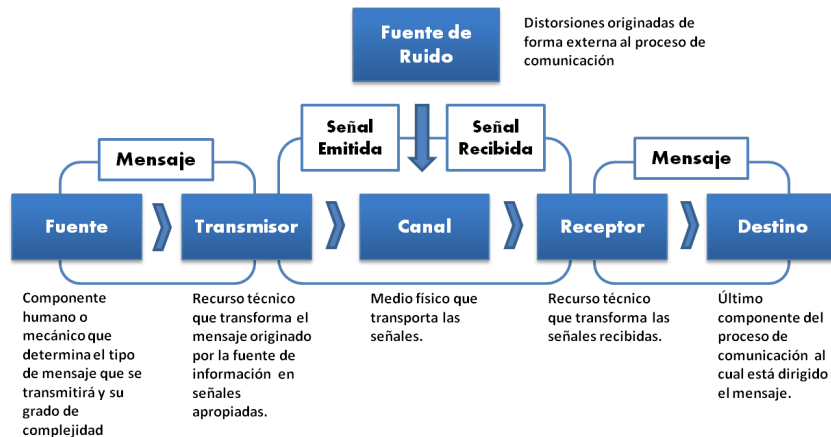


Figura 1.1: *Representación del modelo de Comunicación*

- **Mensaje:** Conjunto de señales, signos o símbolos que son objeto de una comunicación. Contenido de esta comunicación.
- **Canal:** Es el conducto físico por el que se transmite el mensaje.
- **Código:** Sistema de signos y de reglas que permite formular y comprender el mensaje.
- **Receptor:** Es quien recibe el mensaje.
- **Contexto:** Es la situación o entorno extralingüístico en el que se desarrolla el acto comunicativo.

En la figura 1.1 se muestra un esquema del flujo del modelo de comunicación antes expuesto.

1.2. Canal de comunicación

El canal de comunicación es el medio físico por el cual la información es transmitida. El conocimiento de las características de dicho canal es fundamental para la comprensión de la estructura y de los procesos que se dan en un sistema de telecomunicaciones. En la actualidad se utilizan distintos tipos de medios de transmisión dentro de los cuales se incluyen cables coaxiales, fibra óptica y el espacio libre.

Debido a la alta demanda de movilidad el medio radio ha experimentado un notable incremento en su uso. A continuación se describen los efectos que puede sufrir la señal transmitida por este canal.

1.2.1. Atenuación

Cuando una señal se propaga a través de un canal radio esta sufre una pérdida de potencia, esto se debe principalmente a que el frente de onda crece de forma esférica reduciendo así la densidad de potencia de manera proporcional al área de la misma. Al aumentar la distancia desde el punto de emisión (centro de la esfera), el área de la esfera aumenta de forma cuadrática ($4\pi r^2$) por lo que la densidad de potencia también disminuye en dicha proporción, este tipo de atenuación está presente siempre, y se conoce como pérdida de espacio libre. Se define entonces las pérdidas básicas de propagación en condiciones de espacio libre como el cociente entre la potencia radiada por la antena transmisora y la captada por la receptora, estas pérdidas se deben a que la onda electromagnética al propagarse se atenúa según la ley de la inversa de la distancia como se muestra en la ecuación 1.1.

$$L_f = \frac{\lambda^2}{4\pi d^2} \quad (1.1)$$

Existen otros factores que producen atenuación de la potencia de la señal, como por ejemplo obstáculos entre el transmisor y el receptor que no permiten tener una visión directa entre ellos, y multitrayectos provocados por diversas versiones de la señal en otros instantes de tiempo (reflexiones).

1.2.2. Desvanecimiento por multitrayecto

El multitrayecto es un fenómeno físico que consiste en la propagación de una onda por varios caminos diferentes. Cuando una señal es transmitida por un medio radio, esta está sujeta a los efectos de la reflexión, refracción y difracción. Puede existir un camino directo por el cual la señal viaja sin sufrir ningún tipo de obstrucción, pero en la mayoría de los casos, algunas componentes de esta señal son reflejadas por la tierra y objetos tales como edificios, montañas y vehículos presentes entre el transmisor y el receptor ó son refractadas por diferentes capas atmosféricas. Estas componentes viajan por caminos diferentes y se mezclan en el receptor, cada uno de los trayectos tiene una longitud física diferente de modo que la señal sufre retrasos de transmisión. La superposición de todas las señales en el receptor puede producir interferencias constructivas o destructivas en función de los retrasos involucrados.

El hecho de que el entorno sea cambiante (vehículos en movimiento) en el transcurso del tiempo genera una variación en la señal, a esto se le llama *variación temporal*. Por tanto, un



Figura 1.2: *Representación del multitrayecto*

canal de comunicación radio es variante en el tiempo y presenta desvanecimientos debido a los múltiples trayectos. En la figura 1.2 se puede apreciar la caracterización de este escenario.

El modelo matemático del multitrayecto se puede representar utilizando el método de la respuesta al impulso. La expresión de un canal de este tipo se describe en la ecuación 1.2.

$$y(t) = H(t) = \sum_{n=0}^{N-1} \rho_n e^{j\Phi_n} \delta(t - \tau_n) \quad (1.2)$$

Donde N es el número de impulsos recibidos (equivalente al número de caminos electromagnéticos), τ_n es el retardo temporal del n -ésimo impulso y $\rho_n e^{j\Phi_n}$ representa la amplitud compleja (magnitud y fase) del pulso recibido.

En general, la respuesta al impulso es variante en el tiempo y por esto los factores de retardo, amplitud y fase son función del tiempo.

1.2.3. Desvanecimiento selectivo en frecuencia

El funcionamiento de los enlaces radio con visibilidad directa puede verse seriamente afectado por el desvanecimiento selectivo de frecuencia debido a las distorsiones de amplitud y fase. Este desvanecimiento multitrayecto (o selectivo) puede deberse a las reflexiones que ocurren sobre la superficie terrestre, o puede ser inducido por anomalías atmosféricas como fuertes gradientes de conducción.

Se puede asumir un canal con desvanecimiento selectivo como un filtro cuya respuesta en frecuencia no es constante a lo largo del ancho de banda del mismo. En este tipo de canales se habla de ancho de banda de coherencia para referirse al rango de frecuencias en el cual la función de transferencia del canal se mantiene constante, el tamaño de este rango depende del tiempo de retardo (*delay spread*) de las señales multitrayecto, a medida que el retardo aumenta, el ancho de banda de coherencia disminuye, es decir, su relación es inversamente proporcional. Si una señal tiene un ancho de banda menor al ancho de banda de coherencia se considera al canal como no selectivo, en caso contrario se dice que el canal es selectivo en frecuencia.

Como se ha mencionado antes, la respuesta impulsiva del canal varía con el tiempo, si esta variación se da en un periodo menor al tiempo de duración del símbolo que se transmite se dice que el canal presenta un desvanecimiento rápido (*fast fading*), en caso de que el cambio se diera en un tiempo mayor a la duración del símbolo se estaría en presencia de un desvanecimiento lento (*slow fading*).

En resumen, en desvanecimiento plano, el ancho de banda de coherencia del canal es mayor que el ancho de banda de la señal y por tanto, todas las componentes de frecuencia de la señal experimentarán el mismo nivel de atenuación. En un desvanecimiento selectivo en frecuencia, el ancho de banda de coherencia del canal es menor al ancho de banda de la señal y diferentes componentes de frecuencia de esta experimentarán un desvanecimiento decorrelado.

Los impactos del desvanecimiento producidos por el multitrayecto en canales radio son:

- Reducción de la relación señal a ruido (**SNR**) y en consecuencia incremento de la relación de error de bit (**BER**).
- Reducción de la relación portadora/interferencia (**CIR**) y en consecuencia aumento de la BER.
- Distorsión de la forma de onda del pulso digital, provocando una mayor interferencia entre símbolos y una BER mayor.
- Introducción de diafonía entre las dos portadoras ortogonales (componente I y Q), incrementándose la BER.

1.3. Procesamiento digital de la señal

Para desarrollar este concepto veamos el significado de cada una de las tres palabras que lo componen:

Procesamiento: El procesamiento es la aplicación sistemática de una serie de operaciones sobre un conjunto de datos, generalmente por medio de máquinas, para explotar la información que estos datos representan [9]. Estas acciones en la mayoría de las aplicaciones requieren una gran cantidad de operaciones por segundo.

Digital: Tradicionalmente, lo digital está relacionado con el uso de los números dígitos y en particular con los instrumentos de medida que la expresan con ellos [9]. Las señales digitales son pues series de números que aproximan su contraparte analógica en puntos discretos en el tiempo y el espacio. El proceso de conversión de señales que están presentes en el mundo real, es decir, señales analógicas, a formato digital; recibe el nombre de digitalización.

Cuando digitalizamos la información, la música, la voz o el video podemos manipular estos datos, preservarlos y regenerarlos perfectamente a través de medios electrónicos. Cualquier copia de un archivo digital es exactamente igual a su original, por tanto, este es el principal aporte del procesamiento digital.

Señal: Cantidad o impulso físicamente detectable sobre la cual podemos transmitir información, por ejemplo, una señal de humo, una corriente eléctrica o una señal en código Morse. En telecomunicaciones se distinguen dos tipos de señal: analógica y digital.

De esta forma, el procesamiento digital de señal se puede definir como la aplicación de operaciones matemáticas a señales representadas de forma digital.

1.4. Modulación

La modulación es un concepto básico en las telecomunicaciones, esta consiste en la variación de una propiedad o un parámetro de cualquier señal (portadora), en función de otra señal también llamada moduladora. Las portadoras suelen ser señales armónicas con frecuencias superiores a las de la señal de información, con las cuales se logra trasladar el espectro a frecuencias más altas que permitan su irradiación por medio de antenas de tamaño realizable físicamente.

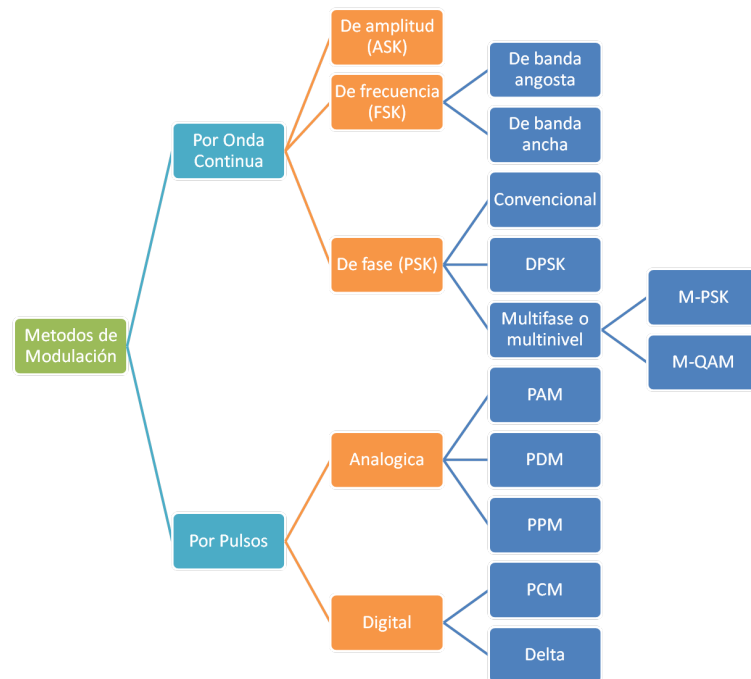


Figura 1.3: *Clasificación de las formas de modulación*

1.4.1. ¿Por qué modulamos?

Al modular se dota a la portadora de rasgos que caracterizan al mensaje que se desea transmitir. A continuación se exponen las principales razones por las que se realiza este proceso:

- La modulación permite transportar la señal a través de un medio físico (aire, cable coaxial, fibra óptica, etc.) a larga distancia.
- Permite utilizar un mismo canal para transmitir varias señales, modulando sobre portadoras de distinta frecuencia (multiplexación), por ejemplo **OFDM**.
- Permite superar limitaciones tecnológicas (ancho de banda relativo).
- Permite la radiodifusión (*Broadcasting*).

Según la naturaleza de la portadora (analógica o digital), podemos clasificar las diferentes formas de modulación en dos grandes grupos, modulación por onda continua y modulación por pulsos. En la figura 1.3 vemos con más detalle dicha clasificación.

1.5. Filtro digital

Un filtro digital es un sistema que realiza operaciones matemáticas sobre una señal muestreada en tiempo discreto para reducir o mejorar algunos aspectos de dicha señal. Generalmente este sistema está compuesto por un conversor analógico-digital que toma muestras de la señal de entrada, seguido por un microprocesador y algunos componentes periféricos como memoria para el almacenamiento de datos y de los coeficientes del filtro. Por último, un conversor digital-analógico completa el proceso a la salida.

Los filtros digitales se caracterizan por su función de transferencia o lo que es lo mismo, su ecuación diferencial. Un análisis matemático de esta función de transferencia permite describir cómo será su respuesta ante cualquier entrada. La ecuación 1.3 representa la forma de un filtro recursivo con los datos de entrada en el numerador y los datos de salida en el denominador, lo que típicamente se conoce como un filtro de respuesta al impulso infinita (IIR), pero si se hace el denominador igual a la unidad, es decir, no hay realimentación, entonces se habla de un filtro de respuesta al impulso finita (FIR).

$$H(z) = \frac{B(z)}{A(z)} = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} \dots + b_N z^{-N}}{1 + a_0 + a_1 z^{-1} + a_2 z^{-2} \dots + a_M z^{-m}} \quad (1.3)$$

El filtro digital es una parte muy importante en el procesamiento digital de señales y está presente en un gran número de aplicaciones. A continuación se nombran algunas ventajas y desventajas de este tipo de filtro frente al filtro analógico:

- Puede tener características que son imposibles de conseguir con un filtro analógico, como por ejemplo, una respuesta de fase exactamente lineal.
- La frecuencia de Nyquist, que fija el ancho de banda útil del filtro, queda definida por los tiempos de conversión del Analog to Digital Converter (ADC) y del Digital to Analog Converter (DAC), la velocidad del procesador, la cantidad de operaciones a ejecutar por unidad de tiempo, etc.
- El comportamiento del filtro digital no varía con las condiciones de su entorno (temperatura, humedad, etc), por lo que no es necesaria una calibración periódica.
- Los avances en las técnicas de integración VLSI hacen que sea posible fabricar filtros digitales pequeños, de bajo consumo, y muy baratos.

- El tiempo de diseño y desarrollo de un filtro digital, en especial el diseño del hardware, puede ser muy superior al requerido para el diseño de un filtro analógico.
- Los filtros digitales pueden utilizarse a frecuencias muy bajas y pueden trabajar sobre un amplio rango de frecuencias simplemente cambiando la frecuencia de muestreo.

1.5.1. Filtro IIR

Filtro cuya respuesta al impulso es infinita, es decir, habrá un número infinito de términos no nulos a la salida. Dicha salida es función de las entradas actuales, pasadas y además, de las salidas en instantes anteriores.

$$y[n] = \sum_{k=0}^{N-1} b_k x[n-k] - \sum_{k=1}^{M-1} a_k y[n-k] \quad (1.4)$$

Características principales de este tipo de filtro:

- Presenta las mismas prestaciones que un filtro FIR con menor orden del filtro.
- Puede ser inestable.
- La fase no es lineal.

1.5.2. Filtro FIR

Filtro cuya respuesta al impulso es finita, es decir, habrá un número finito de términos no nulos a la salida. Dicha salida es función de las entradas actuales y pasadas. A diferencia del filtro IIR este no depende de datos de salida en instantes de tiempo anteriores.

$$y[n] = \sum_{k=0}^{N-1} b_k x[n-k] \quad (1.5)$$

Características principales de este tipo de filtro:

- Se puede diseñar tal que su fase sea lineal, lo cual hace que presente ciertas propiedades en la simetría de los coeficientes.
- Es siempre estable, todos sus polos están en el origen.
- Necesita un orden mayor respecto a los filtros IIR para cumplir las mismas prestaciones. Esto se traduce en un mayor gasto computacional.

1.5.3. IIR frente a FIR

Es muy importante conocer los motivos por los que se debería elegir entre uno de estos dos filtros en la aplicación que se desea implementar, para ello es necesario analizar las diferencias entre ambos filtros y sus principales características que se enumeran a continuación:

1. Los filtros FIR se pueden diseñar para que tengan una fase estrictamente lineal. No sucede lo mismo con los filtros IIR cuya fase es no lineal, especialmente cerca de la zona de transición.
2. Los filtros FIR implementados de forma no recursiva son inherentemente estables. Por el contrario, los filtros IIR por ser sistemas realimentados pueden ser no estables.
3. Los efectos causados por los errores de redondeo en las operaciones aritméticas así como los errores de cuantificación de los coeficientes tienen mayor repercusión en los filtros IIR que en los FIR.
4. Los filtros FIR necesitan un mayor número de coeficientes que los filtros IIR para cumplir las mismas especificaciones. En consecuencia, los requerimientos de memoria, el número de operaciones y los tiempos de procesamiento son mayores para los primeros.
5. Se puede realizar un equivalente analógico con los filtros IIR que satisfaga las especificaciones de diseño. Esto no es posible con un FIR porque carece de una contraparte analógica.
6. Es más sencillo sintetizar filtros con respuestas en frecuencia arbitrarias utilizando filtros FIR.

Por todo lo anterior, se puede decir que para requerimientos de bandas de transición estrechas (caídas abruptas) y eficiencia de cómputo es más adecuado un filtro IIR. Para aplicaciones que exijan una distorsión de fase baja o nula y transiciones no abruptas (número de coeficientes no muy elevado) es recomendable un filtro FIR.

Capítulo 2

Entorno de Trabajo

En este capítulo se describe las herramientas, software y hardware, utilizadas para la realización de este proyecto. Se empezará con una breve pero concisa descripción de los programas empleados en la síntesis e implementación del código programable y a continuación se hará un estudio más detallado de los componentes físicos de la placa.

2.1. Herramientas Software

El dispositivo **Software Defined Radio (SDR)** está integrado por dos módulos principales (un procesador digital de señales (**DSP**) y un dispositivo semiconductor programable (**FPGA**)) que tendrán que ser tratados por dos plataformas distintas. Para la programación y configuración del módulo DSP se trabaja en lenguaje C/C++ o código ensamblador y para el desarrollo del código programable en la FPGA se utiliza el lenguaje **VHDL**.

2.1.1. Code Composer Studio

El desarrollo de las configuraciones sobre la DSP se ha llevado a cabo con el *Code Composer Studio 3.3*. Esta herramienta viene incluida en el paquete de desarrollo software de la placa (**BSDK**). El **CCS** es un conjunto de procedimientos desarrollado para un hardware específico como lo es el procesador digital de *Texas Instrument*, presenta un ambiente de tipo *Visual Studio* y permite operar con comandos propios del lenguaje ANSI C. Además, permite la posibilidad de realizar análisis en tiempo real a través de herramientas como el DSP BIOS. Para más información sobre esta plataforma consultar el documento [14].

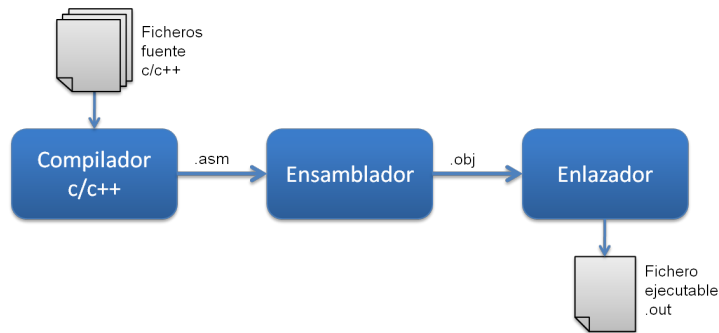


Figura 2.1: *Flujo de desarrollo del código en c/c++*

En la figura 2.1 podemos ver el proceso de desarrollo del código que finalmente servirá para configurar los distintos módulos del dispositivo SDR. Se empezará con la creación de un proyecto en el cual se incluirán todos los ficheros fuentes que contienen el código de configuración, la interfaz gráfica presenta un botón *Built All* que permite compilar, ensamblar y enlazar todo en un solo click. Si no hay errores se generará un fichero ejecutable con extensión .out.

2.1.2. Xilinx ISE Foundation

En la síntesis del diseño digital se ha empleado el programa *Xilinx ISE Foundation 9.2i*, esta plataforma de desarrollo consiste en un conjunto integrado de herramientas software y hardware que permiten implementar y simular diseños digitales en una FPGA. Este programa está dotado de una interfaz gráfica orientada al usuario que facilita el uso de todas sus aplicaciones por medio de iconos, menús y barras de herramientas. Para más información sobre esta plataforma consultar [16].

El proceso de diseño de un módulo hardware sobre un array de puertas programable se muestra en la figura 2.2, en ella se puede ver las tres fases en las que se divide. Como primer paso tenemos la creación del modelo del componente que se pretende realizar, para ello en un editor de texto se describe el diseño en lenguaje VHDL. Una vez concluido éste, se puede verificar su comportamiento mediante simulación. Posteriormente, se pasa a la síntesis que como resultado genera un fichero con extensión **EDIF** ó **XNF** que contiene el listado de conexiones. Si en este paso se fijan restricciones de diseño (topológicas o temporales) que afectan al proceso de implementación éstas se exportan junto con la lista de conexiones en un fichero **NCF**. Por último, se procede a la implementación del diseño, en el Apéndice B se describe con más detalle este paso. Como resultado la herramienta BITGEN genera un fichero con extensión .BIT para la

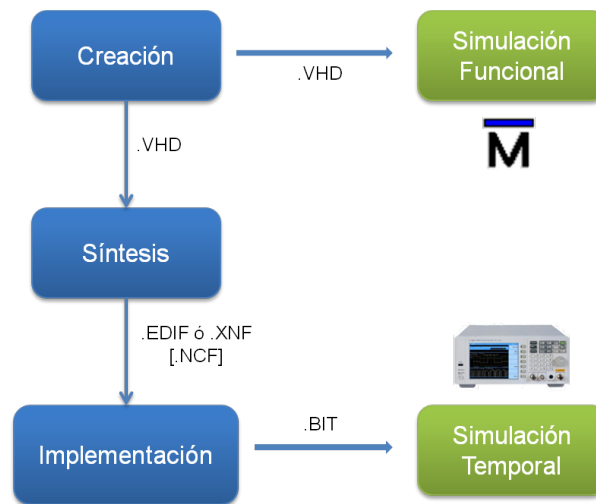


Figura 2.2: Flujo de desarrollo hardware

configuración de la FPGA. Para comprobar el comportamiento final se realizan pruebas mediante simulación temporal cuyos resultados los podemos ver en un osciloscopio.

2.1.3. Otros programas

En la simulación del código desarrollado para el sistema de prueba se ha utilizado *Modelsim* que combina la tecnología de un simulador de un solo núcleo con un entorno de depuración unificada para lenguajes como VHDL, permitiendo de esta forma el análisis y la comprobación de la mejor solución para el diseño implementado en la placa.

Se ha utilizado *Matlab* para el desarrollo del prototipo de demodulador IQ que ha permitido obtener una versión preliminar de su comportamiento. *Matlab* es un lenguaje de computación técnica de alto nivel y un entorno interactivo para desarrollo de algoritmos, visualización de datos, análisis de datos y cálculo numérico.

2.2. Herramientas Hardware

En este apartado se describen los elementos físicos utilizados en el proyecto, sus características técnicas más relevantes así como las funciones que desempeñan en el cumplimiento de nuestro propósito. Para información más detallada sobre los componentes del dispositivo consultar el manual del fabricante [8].

Como herramienta física se ha utilizado la plataforma de desarrollo SFF SDR (*Small Form*

Factor Software-defined-radio), fabricada por la compañía Canadiense Lyrtech, fundada en 1983, esta empresa es líder en productos de procesamiento digital de señales.

En un sistema **SDR** el transmisor genera las formas de onda como señales digitales muestreadas, convertidas de digital a analógicas a través de un **DAC** de banda ancha, posteriormente traspone la señal de frecuencia intermedia f_{IF} hacia una frecuencia más alta o radiofrecuencia f_{RF} . De forma similar, el receptor desplaza la señal recibida a f_{RF} a la frecuencia intermedia f_{IF} , emplea un conversor analógico-digital **ADC** de banda ancha que captura toda la información disponible en los canales del nodo radio y finalmente demodula las formas de onda utilizando el software implementado sobre la **FPGA**.

Por todo lo anterior, la plataforma está constituida por tres placas, la primera de ellas empezando por la base se encuentra el módulo de procesamiento digital, en la parte central está ensamblado el módulo de conversión de datos y en la parte superior el módulo de radiofrecuencia (ver figura 2.3). A continuación se describe las partes principales de cada uno de estos módulos.

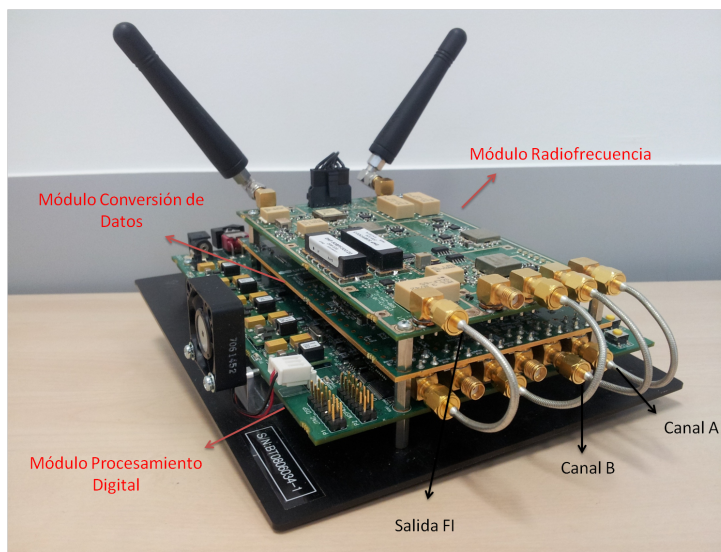


Figura 2.3: *Plataforma SFF SDR.*

2.2.1. Módulo de procesamiento digital

Esta placa contiene dos componentes principales:

- **Procesador digital:** Utiliza el chip integrado TMS320DM6446 DMP de Texas Instrument, que cuenta con procesamiento de video, tiene acceso a memoria DDR2 SDRAM y a otros periféricos útiles en un entorno de sistema. La arquitectura de doble núcleo que posee este

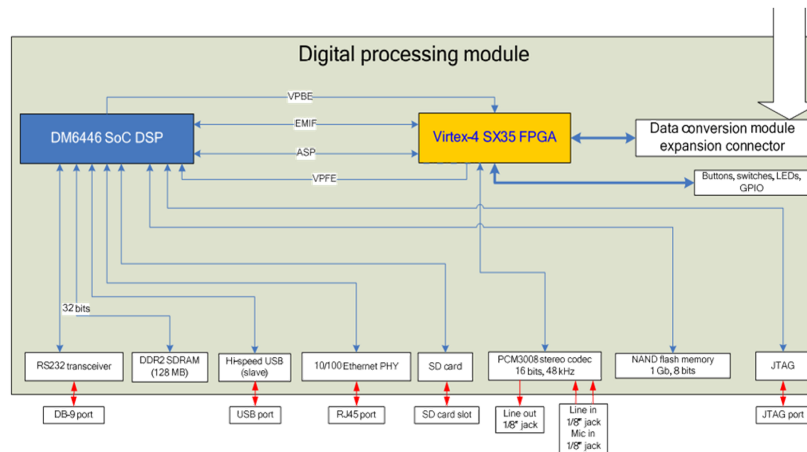


Figura 2.4: *Bloques del módulo DSP.*

componente ofrece los beneficios de la tecnología DSP y RISC (Reduced Instruction Set Computer) incorporando los núcleos TMS320C64x+ DSP y ARM926EJ-S. La transmisión de datos entre el DMP y la FPGA se hace a través de los puertos **EMIF** y **VPSS**.

- **FPGA:** Esta sección está equipada con una **FPGA** Virtex-4 XC4SX35 de Xilinx, cumple la función de coprocesador y sirve como interfaz para todos los dispositivos de Entrada/Salida integrados. Entre sus principales características cuenta con 34.560 celdas lógicas, un array de 96x40 bloques de celdas lógicas y 3.456 kilobits de RAM. Los registros de este dispositivo son utilizados para programar las diferentes interfaces de la FPGA sobre el módulo de procesamiento digital. Se puede acceder a dichos registros a través del DSP o de un dispositivo host. Por tanto, tenemos dos formas de interactuar con la FPGA, la primera es a través de la DSP, con ayuda del DSP **API** [7] podemos establecer la configuración de los módulos de conversión de datos y de radiofrecuencia; la segunda forma es a través de la consola de comandos de la plataforma de desarrollo de Lyrtech *smshell* [6], por medio de esta herramienta podemos cargar un fichero de configuración (.bit) en la placa.

En la figura 2.4 podemos ver el conexionado de todas las partes que componen este módulo, así como los protocolos y las interfaces que están involucradas. En la parte inferior se encuentran todos los periféricos que permiten la conexión con el exterior y en la parte superior derecha se encuentra el conector de expansión que sirve de puente para la comunicación con el módulo de conversión de datos.

2.2.2. Módulo de conversión de datos

En la figura 2.5 vemos el diagrama de bloques de este módulo, en la parte inferior se observa el bus que comunica éste con el DSP y en la parte superior izquierda los amplificadores programables se comunican con el módulo de radiofrecuencia. A continuación se describen los componentes más relevantes:

- **FPGA:** Este componente proviene de la serie Virtex 4 LX, se utiliza para controlar el DAC, los dos ADC y la adquisición de datos. Se comunica con el módulo DSP a través del conector de expansión. Cuenta con 24.192 celdas lógicas, un array de 96x28 bloques de celdas lógicas y 1.296 kilobits de RAM.
- **DAC:** El integrado DAC5687 [13] es un conversor Digital-Analógico de alta velocidad (500 MSPS), con 16 bits de resolución y doble canal, cuenta con filtros de interpolación, un oscilador complejo controlado numéricamente (NCO) y compensación IQ. Permite diferentes modos de operación, entre estos se encuentra el modo *Dual-channel*, donde las salidas del DAC son independientes, que facilita el trabajo con señales banda base con modulación en cuadratura.
- **ADC:** Utiliza el integrado ADS5500 con 14 bits de resolución, lo que permite una tasa de muestreo de hasta 125 MSPS. Además, esta sección cuenta con un atenuador digital y un amplificador programable, la combinación de estos dos elementos permite obtener diferentes ganancias para distintas señales de entrada. El atenuador digital, el semiconductor PE4305, se utiliza para disminuir la amplitud de la señal de entrada hasta 15.5 dB en incrementos de 0.5 dB. El amplificador programable es el chip LT5514 y se utiliza para proporcionar una ganancia en la amplitud sobre un rango de 22.5 dB. De esta forma, cuando combinamos estos dos elementos podemos conseguir una ganancia de -8 dB a 30 dB.

2.2.3. Módulo de radiofrecuencia

En esta sección se proporciona una descripción de las principales características del módulo de RF así como una ligera explicación del funcionamiento del mismo.

Transmisor: Permite la transmisión de señales de radiofrecuencia en dos bandas (262-438 MHz y 523-876 MHz), entre sus principales componentes se destacan:

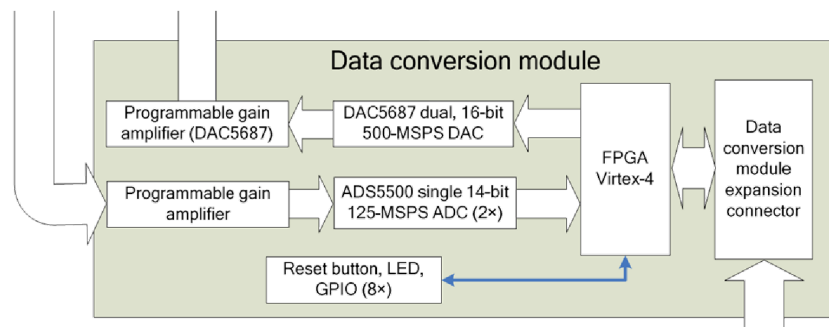


Figura 2.5: Bloques del módulo de conversión de datos.

- **Oscilador local:** El oscilador local del transmisor lo componen el PLL TRF3701, un oscilador controlado por tensión (VCO) y un pre-escalador que divide por dos, este último componente puede ser activado o desactivado en función de la banda en la que se desee transmitir. Es decir, cuando el pre-escalador está activado, tendremos a la salida del **VCO** una frecuencia que puede variar entre 262 MHz y 438 MHz. Cuando dicho elemento está desactivado, el rango de frecuencia a la salida es de 523 MHz hasta 876 MHz.
- **Modulador en cuadratura:** Es el componente TRF3701 de *Texas Instrument* que se utiliza para obtener una señal de banda lateral única a la salida, convierte la señal de frecuencia intermedia (menor de 65 MHz debido al filtro paso bajo que hay en la entrada) a RF.
- **Reloj de referencia:** Este reloj se utiliza tanto en el transmisor como en el receptor para asegurar que todas las frecuencias están en fase. Existen dos opciones posibles de configuración, la primera de ellas es un reloj de referencia interno que trabaja a 10 MHz, la segunda es un reloj externo que se puede conectar a través del conector de reloj de referencia del módulo RF.

Receptor: Esta sección está formada por un receptor heterodino de tres etapas que consigue una frecuencia intermedia final de 30 MHz y un filtro paso banda de ancho de banda seleccionable entre 5 o 20 MHz en función de la aplicación desarrollada. Veamos sus principales características:

- **Filtros:** Esta sección contiene en total 4 filtros cuyo principal objetivo es rechazar el ruido existente fuera de la banda de interés. El primero de ellos es un FPB con frecuencia de corte 1 GHz. El segundo es un filtro paso banda, de banda estrecha (20 MHz), centrado en

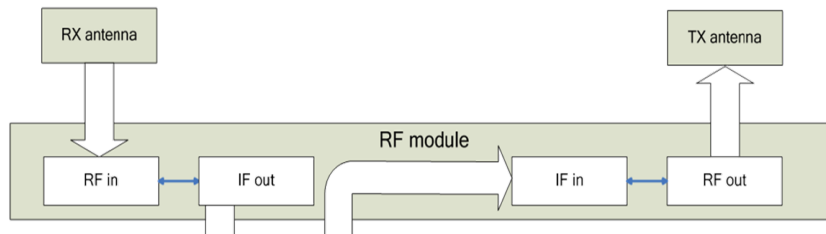


Figura 2.6: *Bloques del módulo de Radiofrecuencia.*

1.575 MHz. El tercero es también un filtro paso banda de banda seleccionable entre 5 o 20 MHz como se comentó anteriormente y centrado en 300 MHz. Por último, un filtro paso bajo con frecuencia de corte 65 MHz que permite obtener la señal de frecuencia intermedia (30 MHz) que se envía al módulo de conversión de datos.

- **Oscilador local:** El receptor utiliza tres osciladores locales para llevar la señal de radiofrecuencia a frecuencia intermedia de tal forma que la información pueda ser procesada por el módulo de conversión de datos. El primer OL incorpora un pre-escalador que permite obtener frecuencias entre 1,6 y 2,5 GHz. El segundo oscilador está fijado a 1.275 MHz, mientras que el tercer y último OL está fijado a 330 MHz.
- **Mezcladores:** Se utilizan para mezclar la señal del oscilador local con la señal recibida y llevarlas a la frecuencia intermedia deseada. Hay tres mezcladores en este módulo, uno para cada OL. El primer mezclador se utiliza para mezclar la señal proveniente del RX-LO1 con la señal filtrada por el FPB de 1 GHz llevando la LSB a un filtro paso banda de ancho de banda 20 MHz (entre 1,575 y 1,585 GHz). El segundo mezclador a 1,275 GHz se utiliza para llevar la **LSB** de 1,575 GHz a 300 MHz, a través del filtro de banda seleccionable (5 MHz/20 MHz). Finalmente, el tercer mezclador a 330 MHz se utiliza para llevar la LSB de la señal a 30 MHz.
- **Antenas:** El dispositivo está equipado con dos antenas que trabajan en las bandas **GMRS/FRS** y ofrecen un compromiso entre tamaño y potencia. Estas tienen una ganancia aproximada de 2.5 dBi.

En la figura 2.6 vemos una representación de los bloques que conforma el módulo de radiofrecuencia. En la parte inferior se puede observar los buses que conectan a este con el módulo de conversión de datos.

Capítulo 3

Modelo de Prueba

En este capítulo se discuten los motivos por los cuales se decidió implementar un sistema **OFDM** así como los objetivos que se pretenden conseguir con este desarrollo. Posteriormente, se presenta una explicación del modelo físico del mismo para entender mejor los pasos de diseño y las dificultades que se pudieron encontrar durante su implementación. Finalmente, se muestra un resumen del trabajo realizado en otras fases del proyecto, las modificaciones llevadas a cabo para conseguir mejor rendimiento y gestión de los recursos disponibles y la descripción de nuevos módulos.

3.1. ¿Que se va a implementar?

Dentro de los principales campos de aplicación comercial de las plataformas **SDR** se encuentran los lectores **RFID**, equipos terminales (PCE) **Wimax** ó Wi-Fi, sistemas de datos de banda ancha, entre otros. Muchas de estas aplicaciones utilizan como técnica de transmisión de datos la Multiplexación por División de Frecuencia Ortogonal (OFDM) debido a su eficiencia espectral y a su buen comportamiento en canales radio con multitrayecto.

Gracias a la alta capacidad de procesamiento de los circuitos actuales esta técnica, que se inicio en 1970, ha experimentado un auge en su implementación y sistemas como **ADSL**, **DVB-T**, telefonía 4G **LTE**, por nombrar algunos, la utilizan. Por esta razón y porque además implementar un modelo OFDM nos permite explorar y probar todos los módulos del dispositivo SFF SDR se decidió trabajar con este modelo.

En este proyecto se tomó como referencia el sistema OFDM propuesto en el documento

Design and Implementation of Synchronization and AGC for OFDM-based WLAN Receivers [4], escrito por cuatro prestigiosos investigadores de la Universidad Carlos III de Madrid. Por tanto, con el desarrollo de este sistema se pretende probar cada uno de los módulos de la plataforma SDR, medir su operatividad verificando sus límites de funcionamiento, establecer un mecanismo de trabajo que permita, de forma sencilla, implementar cualquier otro sistema o técnica que se adapte al soporte físico que tenemos y finalmente, poner en práctica las técnicas de sincronización y control de ganancia propuestos en el documento.

3.2. Modulación OFDM

OFDM es una técnica de transmisión de múltiples portadoras ortogonales entre sí, donde la información es transmitida sobre múltiples subportadoras, esto es, un bloque de N símbolos que son transmitidos en serie en un periodo de símbolo (T_s) cada uno, se convierten en un bloque de N símbolos en paralelo que se transmiten en un tiempo N veces el periodo de símbolo ($T = N.T_s$ segundos), con esto se consigue que el ancho de banda que se ocupaba enviando solo una portadora ahora se ocupa enviando N subportadoras obteniendo así eficiencia espectral. Además, al incrementar el periodo de símbolo se consigue reducir la interferencia entre símbolos.

El mayor inconveniente que presenta este sistema es el requerimiento de una sincronización perfecta tanto en tiempo como en frecuencia. Pero las ventajas conseguidas con OFDM son muchas más, lo que ha hecho que esta técnica de transmisión sea tan popular. Por ejemplo, un canal con desvanecimiento típico afectará solo a unas cuantas subportadoras, en la mayoría de casos este efecto puede ser compensado realizando un espaciado eficaz entre símbolos y codificación de canal. Otra característica importante de OFDM es que se puede implementar de forma muy eficiente haciendo uso de la transformada inversa de Fourier (**IFFT**) en el transmisor y la transformada rápida de Fourier (**FFT**) en el receptor.

Para obtener los mejores resultados con esta técnica, es necesario que las subportadoras estén perfectamente sincronizadas y que dicha sincronización nunca se pierda, pues se perdería todas las prestaciones de este sistema. Idealmente, para mantener la ortogonalidad es necesario que el periodo de símbolo sea exactamente inverso al ancho de banda de las portadoras y que se realice la FFT de tamaño un número entero de ciclos. Además, las portadoras consecutivas deben diferir solo en un ciclo. Si se trabaja en entornos multitrayecto, cada subportadora podría experimentar un desvanecimiento plano, por lo que el ancho de banda de cada subportadora

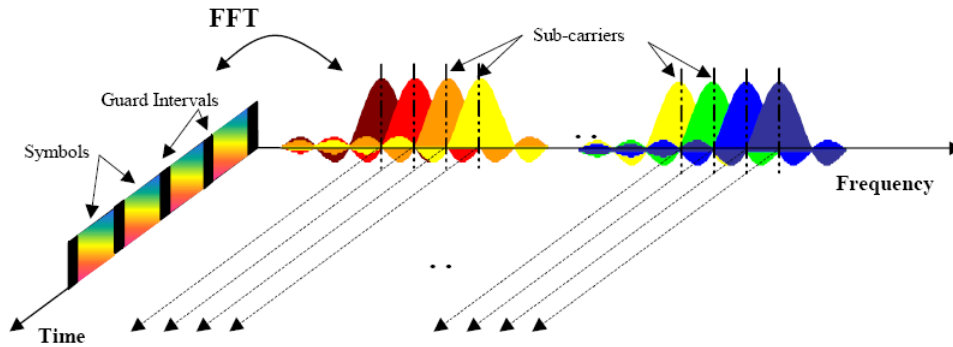


Figura 3.1: *Trama OFDM en tiempo y frecuencia.*

tendrá que ser menor que el ancho de banda de coherencia y cada símbolo podría experimentar un canal invariante en el tiempo, por tanto el periodo de símbolo deberá ser menor que el tiempo de coherencia.

El problema de la interferencia entre símbolos se resuelve implementado un tiempo de guarda, introducido en el dominio del tiempo entre dos símbolos OFDM y realizando la FFT de tamaño tal que no interfieran componentes del símbolo anterior y posterior. Por otra parte, el problema más acuciante en los sistemas multiportadora es la interferencia entre portadoras **ICI**, como resultado de la pérdida de ortogonalidad entre subportadoras, esto ocurre cuando se realiza la FFT en lugares donde la subportadora no está presente (N no es un número entero de ciclos), que podría ser el caso cuando el multitrayecto está presente y el tiempo de guarda es cero. Este efecto se puede reducir por medio del uso de un prefijo cíclico que consiste en copiar las últimas M muestras del símbolo OFDM en su cabecera y de esta forma asegurar la ortogonalidad sobre el periodo de la FFT.

En la figura 3.1 podemos ver una secuencia de símbolos OFDM separados entre sí por un intervalo de guarda, en el dominio de la frecuencia se puede observar la ortogonalidad de cada portadora.

3.3. Descripción del sistema

En esta sección se describirá el modelo de prueba desarrollado:

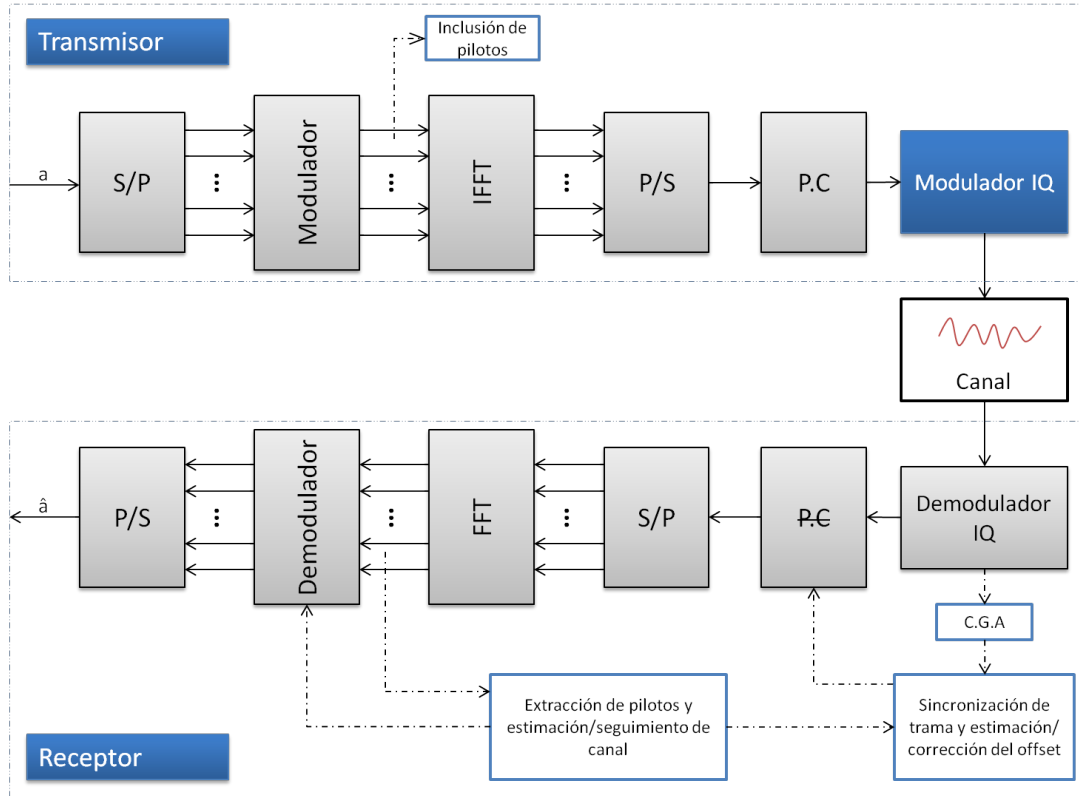


Figura 3.2: Diagrama de Bloques Sistema OFDM.

3.3.1. Estructura

El sistema está compuesto por dos módulos. En el transmisor los datos en forma de flujo de bits son modulados realizando una asignación de símbolos adecuada previa conversión serie-paralelo. Posteriormente, se realiza la transformada inversa de Fourier (IFFT) que permite obtener la señal en el dominio del tiempo, la cual vuelve a un formato serie y antes de ser transmitida se incluye el prefijo cíclico (CP). En el receptor la señal debe ser sincronizada en tiempo y frecuencia y el proceso restante se tiene que llevar a cabo de forma inversa a la realizada en el transmisor. En la figura 3.2 se muestra el diagrama de bloques del sistema, en color gris se resaltan los bloques conseguidos hasta esta fase del proyecto global, en azul se representa el modulador IQ que viene integrado en la herramienta y los cuadros en blanco representan los módulos que pueden ser implementados en ampliaciones futuras.

3.3.2. Modelo de la señal

La envolvente compleja de una señal OFDM viene determinada por la ecuación 3.1.

$$x(t) = \frac{1}{\sqrt{N}} \sum_{s=1}^{N_s} \left[\sum_{k=0}^{N-1} X_{s,k} e^{j2\pi k \Delta f t} \right] g(t - sT') \quad (3.1)$$

Donde :

$X_{s,t}$ Información modulada enviada en la K -ésima subportadora en el t -ésimo símbolo OFDM.

N_s Número total de símbolos OFDM.

$g(t)$ Función con forma de pulso rectangular.

Δf Espaciado en frecuencia.

$T' = T + T_g$ Duración del símbolo.

T_g Duración del prefijo cíclico.

N Número de subportadoras.

La señal en tiempo discreto en el receptor, una vez haya sufrido los efectos del canal, puede escribirse como:

$$r_{s,n} = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} X_{s,k} H_{s,k} e^{j2\pi n \frac{k+\epsilon}{N}} + w_{s,n} \quad (3.2)$$

Donde :

$H_{s,k}$ Respuesta en frecuencia del canal en la k -ésima subportadora para el s -ésimo símbolo OFDM.

w Representa el ruido blanco (AWGN).

3.3.3. Formato de trama

El formato de trama elegido es el definido en el estándar **IEEE** 802.11a para WLAN de alta velocidad. En este modelo cada paquete contiene una cabecera, que se usa para la estimación de canal, la detección y la sincronización de la señal recibida. Este preámbulo se compone de cinco símbolos OFDM conocidos tanto en el transmisor como en el receptor. Los dos primeros símbolos reciben el nombre de *Símbolos de entrenamiento corto* (STS) y su propósito es la detección de la señal, la estabilización a través del control de ganancia (AGC), la diversidad (en caso de haber

varias antenas) y la sincronización en frecuencia. Estos STS están formados por la repetición de diez símbolos cortos de 16 muestras cada uno. La segunda parte del preámbulo está conformada por dos símbolos idénticos y un prefijo cíclico, esta secuencia recibe el nombre de *Símbolos de entrenamiento largo* (LTS) y se utiliza para la estimación del canal y la corrección del offset. El último símbolo tiene por nombre SIGNAL y contiene información relativa al tamaño del paquete y al tipo de modulación utilizada.

3.4. Fase inicial

En la anterior fase de este proyecto se realizó el primer diseño del transmisor OFDM que consta de un modulador QPSK, cuatro bloques de memoria RAM para las conversiones serie-paralelo y paralelo-serie, y un módulo que efectúa la transformada inversa de Fourier para obtener los símbolos OFDM.

Respecto a la señal transmitida, esta se diseñó como una trama de 100 símbolos OFDM, donde:

- Un símbolo OFDM se obtiene de 64 símbolos QPSK que a su vez están compuestos por dos bits.
- A cada símbolo OFDM se le añade en la parte posterior las 8 primeras muestras del símbolo para mitigar los efectos de la ICI.
- A cada trama se le añade la secuencia corta de entrenamiento descrita en el apartado anterior para la detección y estabilización de la señal recibida.

En la figura 3.3 se muestra el material disponible de esta primera fase, esto es, los ficheros con extensión *.vhdl* que describen los componentes del modulador.

3.5. Fase actual

En esta fase del proyecto se han llevado a cabo diversos cambios en diferentes niveles sobre el diseño y la implementación del transmisor OFDM. A continuación, se detallan dichos cambios describiendo cada uno de los niveles sobre los que se actuaron y las repercusiones que estas acciones han tenido en el desarrollo de este trabajo.

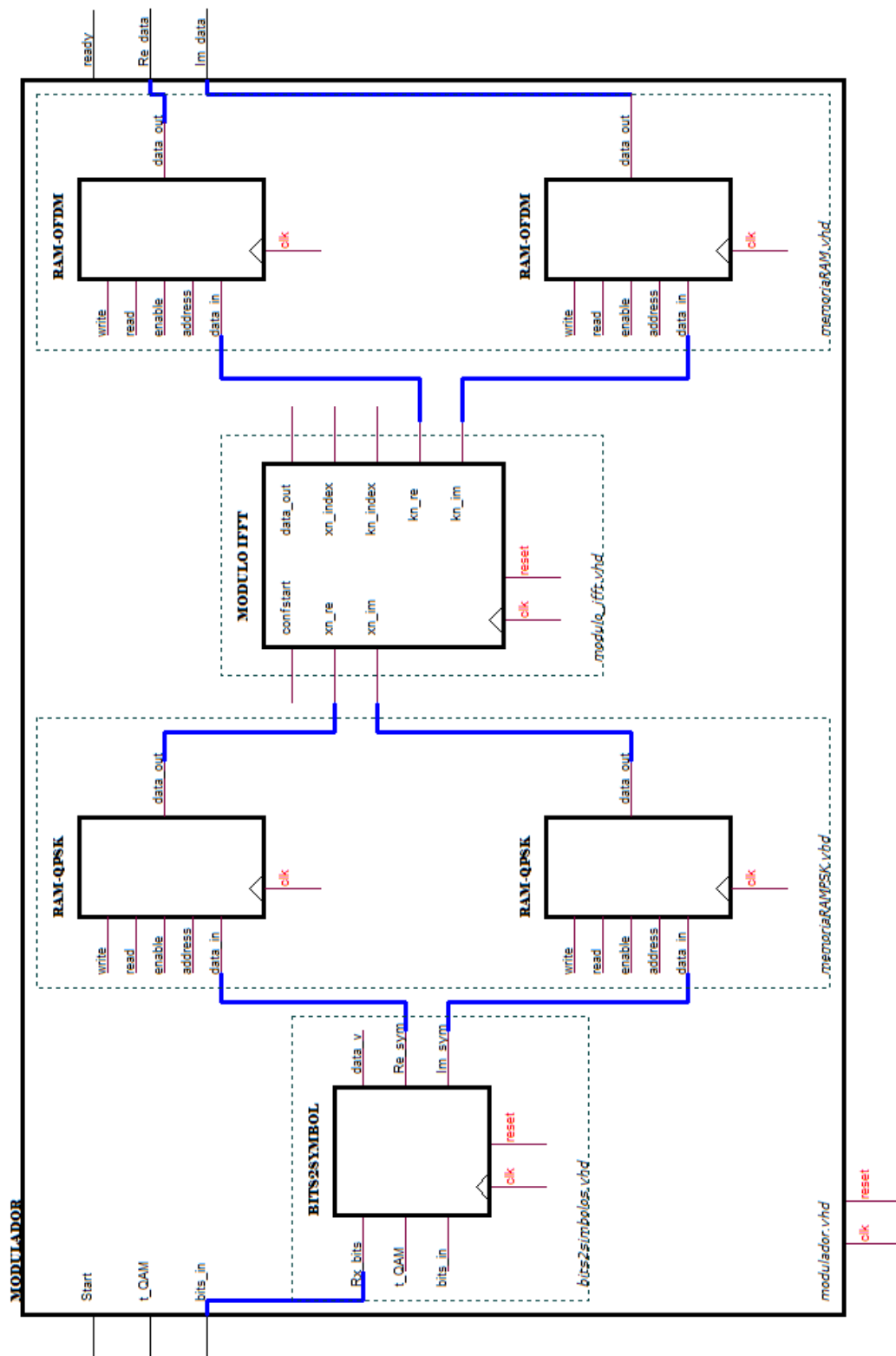


Figura 3.3: Estado Inicial.

3.5.1. Nivel Software

Ante la necesidad de tener un entorno de trabajo propio que permitiese la implementación del sistema OFDM no sólo del transmisor sino también del receptor de forma modular, se creó un proyecto con nombre *OFDM_fase1*, dicho proyecto contiene una serie de carpetas que obedecen a la estructura real de la plataforma SFF SDR como se explicó en el Capítulo 2, es decir, se busca tratar por separado los módulos FPGA y DSP debido a que cada uno de estos módulos utiliza un software de desarrollo diferente. Por tanto, para trabajar sobre la FPGA se creó un Proyecto ISE con nombre *Sistema_OFDM*, la creación y estructuración de este tipo de proyecto se explica en [14]. Para trabajar con el DSP se creó algo similar en el entorno del CCS, este proyecto tiene el mismo nombre que su contenedor. Así pues, la distribución modular ha permitido tener una visión más clara del entorno de trabajo y ha facilitado el desarrollo de cambios sobre el diseño del sistema y la configuración de los distintos módulos de la herramienta de forma independiente.

3.5.2. Nivel Hardware

Como se explicó en el capítulo anterior, la configuración del hardware de la placa se hace a través del DSP (en el CCS) con ayuda de las funciones proporcionadas por el API. El principal logro en este nivel ha sido la configuración del módulo de radiofrecuencia, no obstante ha sido necesario la reconfiguración de parámetros de otros módulos, a continuación se describen los pasos seguidos:

Configuración del módulo de radiofrecuencia: Se comienza con la inicialización del driver del RFFE (Radio Frequency Front End) con la instrucción *RFFE_Init*, a continuación se habilita la alimentación de los módulos transmisor y receptor utilizando *RFFE_SetPowerRX* y *RFFE_SetPowerTX* respectivamente, se selecciona el ancho del filtro en el receptor (seleccionable entre 5 o 20 MHz) con la función *RFFE_SetRxFilter*, esto limita el ancho de banda de las señales que pueden ser recibidas; se selecciona la fuente de reloj de los osciladores *RFFE_SetRefClk*. Por último, se establecen las frecuencias de referencia de los PLL así como las frecuencias de transmisión y recepción deseadas con la función *RFFE_SetBaseFrequency*.

La configuración de este módulo ha permitido elegir una frecuencia de transmisión cualquiera dentro de las posibles en las bandas de transmisión de la herramienta, además, queda de manifiesto que el ancho de banda máximo de transmisión es de 60 MHz mientras que el máximo ancho de banda en recepción es de 20 MHz.

Configuración del módulo de conversión de datos: Este módulo se puede separar en dos partes, por un lado en la parte de transmisión tenemos el módulo DAC y en la parte del receptor el módulo ADC. En la primera fase de este proyecto se realizó una configuración parcial del DAC por lo que aquí sólo se mencionarán los cambios realizados en la configuración de ganancia de los dos canales de este módulo y la parte de amplificación del ADC.

En el DAC se puede configurar la ganancia de cada uno de sus dos canales por separado, para ello el API permite dos grados de libertad en el nivel de dicha ganancia, el primero es un ajuste grueso que puede tomar valores entre 0 y 15, el segundo es un ajuste fino que puede tomar valores entre -128 y 127. Con ayuda de la función *conv_mod_SetDACChanGain* se puede realizar cualquiera de las combinaciones posibles. Esta forma da amplificación permite que la señal transmitida tenga un nivel de amplitud adecuado para mitigar los efectos del multitrayecto.

El procedimiento para la configuración del ADC comparte las mismas instrucciones que para el DAC, en realidad, son instrucciones para configurar el módulo de conversión de datos como un único elemento. Dicho procedimiento fue explicado en la primera fase de este proyecto [15] por lo que aquí se omite. En particular el ADC está conformado por un amplificador y un atenuador, la combinación de estos dos elementos permite conseguir una variación en la ganancia de la señal recibida de -8 dB a 30 dB, la instrucción que permite hacer esto de forma sencilla es *conv_mod_SetADCChanGain*.

3.5.3. Nivel Diseño

En este punto se empezará explicando las modificaciones realizadas sobre el diseño existente de forma ascendente en importancia. Las primeras modificaciones expuestas tienen como objetivo mejorar la estructura del diseño para una mejor compresión y un fácil manejo, las siguientes modificaciones buscan mejorar el comportamiento del sistema.

Como primera medida, se estableció un método sencillo que permitiese cambiar la frecuencia de reloj del transmisor y de esta forma realizar varias pruebas sobre el ancho de banda de la señal transmitida. Posteriormente, se dotó al modulador de una señal de salida que indica si hay datos válidos en su salida. Esta señal sirve de control en la transición del estado *Load* al estado *Transmission* del transmisor.

Se diseñó una entidad nueva llamada transmisor que contiene el modulador, en un nivel superior de jerarquía (es padre del modulador) tiene como fin la transmisión de los datos, por

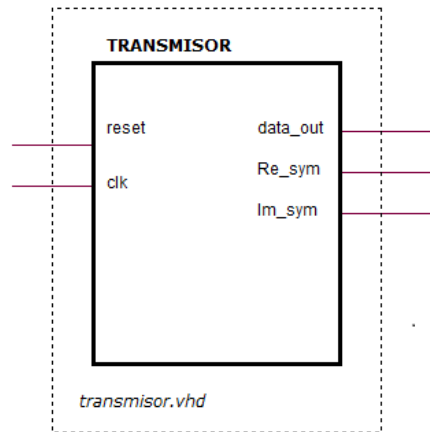


Figura 3.4: *Bloque Transmisor.*

ello será el encargado de hablar con el receptor. En la figura 3.4 podemos ver su diagrama de bloque.

La máquina de estado que gobierna al transmisor contempla tres fases, la primera recibe el nombre de *Start*, en este estado todas las señales de control deben ser inicializadas y además se deben inicializar todos los componentes (es el estado de arranque). Cuando el modulador indica que está listo para recibir datos se pasa al estado *Load*, como su nombre lo indica es la fase de carga de datos que pueden provenir de una fuente aleatoria o bien pueden estar almacenados en memoria como es nuestro caso. Una vez haya finalizado la carga de todos los datos se pasa al estado *Transmission* y se activa la señal que indica que el transmisor está transmitiendo.

Con este nuevo nivel se consigue diferenciar el papel del modulador y el transmisor, dejando las tareas de envío de información al transmisor y que el modulador sólo se ocupe de procesar dicha información. En la figura 3.5 podemos ver la simulación del comportamiento del transmisor en cada una de sus fases.

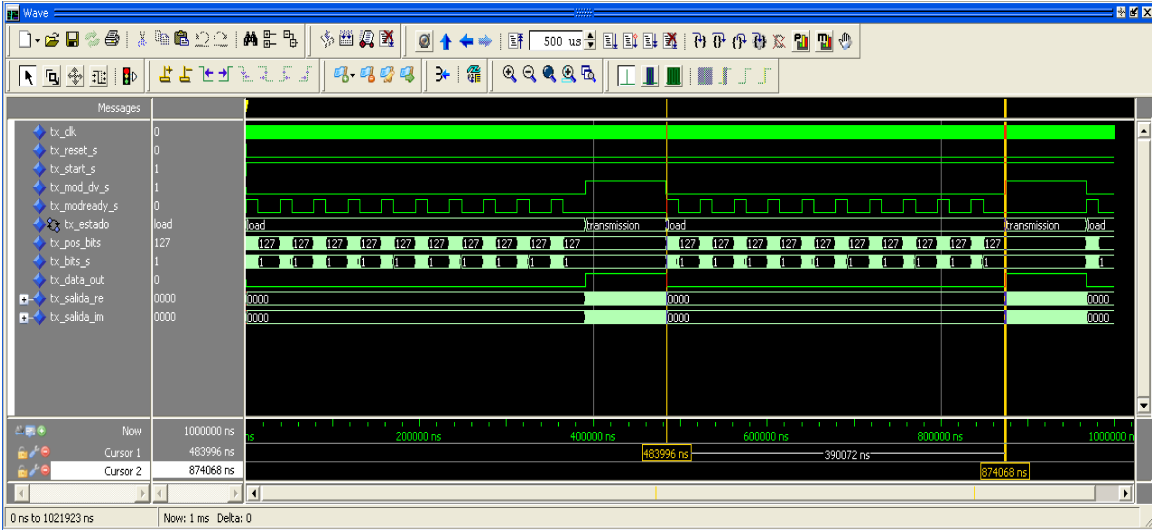


Figura 3.5: *Simulación del Transmisor.*

Demodulador Digital en Cuadratura

En este capítulo se presenta el proceso de diseño y la implementación sobre un dispositivo SDR de un demodulador digital en cuadratura para el receptor del sistema OFDM utilizado en este proyecto. La primera parte plantea el problema de forma teórica para buscar la solución que más se adapte a los recursos disponibles, a continuación se muestra el prototipo realizado para la validación de la elección del diseño y en la parte final se describe su desarrollo en lenguaje de alto nivel para su posterior puesta en marcha sobre el dispositivo radio.

4.1. Motivación

Una vez que se puede transmitir en cualquiera de las frecuencias que están dentro de la banda de transmisión de que dispone la herramienta, se procede a comprobar la recepción de la señal transmitida. Se utilizará el mismo dispositivo como receptor ya que éste cuenta con una antena y un módulo de radiofrecuencia apto para cumplir dicha función. Como se explicó en el capítulo 2 (Herramientas Hardware), el módulo de radiofrecuencia en recepción dispone de sendos osciladores locales que permiten llevar la frecuencia de la señal recibida a f_{RF} a una frecuencia más baja f_{IF} . La utilidad de la frecuencia intermedia radica en el hecho de que todos los circuitos posteriores a esta etapa se podrán diseñar para que trabajen a una frecuencia fija facilitando su diseño.

Supongamos pues que partimos de una señal recibida como la descrita en la ecuación 4.1. Donde $I'(t)$ y $Q'(t)$ serían las componentes en fase y cuadratura respectivamente después de sufrir los efectos del canal. Para aplicar los procesos inversos realizados en transmisión será necesario

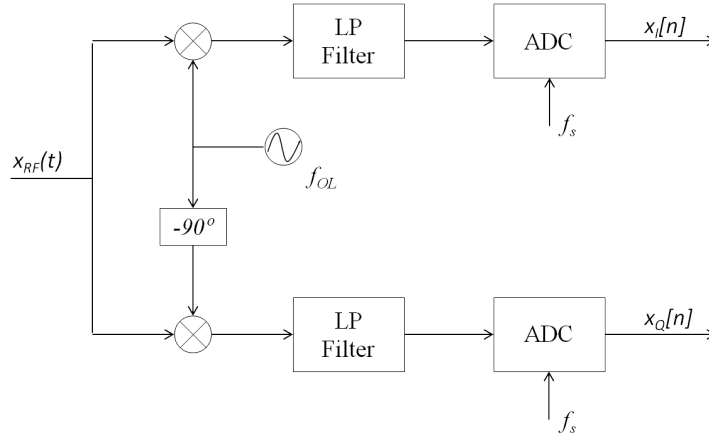


Figura 4.1: Demodulador IQ convencional.

obtener por separado cada componente y para ello se diseñará un demodulador IQ.

$$r(t) = I'(t) \cos(\omega_{RF}t + \phi(t)) - Q'(t) \sin(\omega_{RF}t + \phi(t)) \quad (4.1)$$

Los esquemas de demodulación analógicos convencionales obtienen las componentes en fase y cuadratura (I y Q) por demodulación en banda utilizando dos osciladores cuyas señales están desfasadas $\pi/2$ rad entre sí, el resultado de la mezcla es una señal de frecuencia intermedia por cada componente que posteriormente es filtrada y muestreada individualmente por un ADC [3], este tipo de esquema se muestra en la figura 4.1. Con esta configuración las componentes de los dos canales no están perfectamente adaptadas lo que genera una descompensación entre ellas (introducción de offset y distinta ganancia entre los canales I y Q), produciendo un error de fase de hasta dos o tres grados [11]. Sin embargo, en una demodulación digital directa la señal IF es digitalizada a una tasa superior a la frecuencia de Nyquist y a continuación es demodulada por dos filtros digitales. La descompensación entre los dos canales ya no existe debido a que las muestras I y Q de los canales son procesadas por un mismo componente analógico (el ADC).

4.2. Prototipo

Planteado el problema el siguiente paso es buscar la solución que mejor se adapte a los recursos físicos que se tienen a disposición, esto es, aprovechar que el módulo de conversión de datos cuenta con un potente conversor Analógico-Digital que permite una tasa de muestreo de hasta 125 Msps.

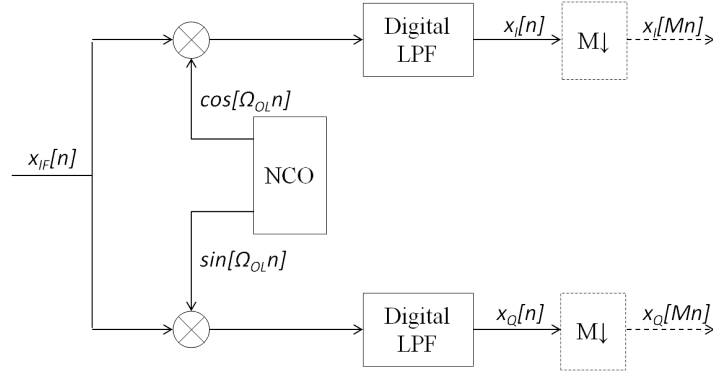


Figura 4.2: Estructura Demodulador IQ Digital.

Como primera opción se planteó realizar la demodulación utilizando la transformada de *Hilbert*, es decir, aplicando un procesamiento digital a la señal recibida y a una versión de esta desplazada $\pi/2$ rad como se explica en [10], pero la longitud de dicha transformada debe ser de un orden considerable, por ejemplo de orden 64, de forma que los errores producidos sean aceptables. Rápidamente se llegó a la conclusión que implementar este modelo en un lenguaje de alto nivel no iba a ser sencillo.

Como segunda opción se pensó realizar el proceso inverso al modulador IQ que contiene el módulo de RF. En este caso, la señal a frecuencia intermedia se lleva por dos caminos paralelos para ser multiplicada por un coseno y un seno, que están a una frecuencia cercana a la señal de entrada, el resultado de esa mezcla es la suma y la diferencia en frecuencia de las dos señales. Cada rama se conecta a un filtro paso bajo para eliminar la componente de mayor frecuencia. En la figura 4.2 se puede ver una representación del proceso descrito. Esta solución que comprende tres tipos de circuitos diferentes se adapta más a los recursos disponibles.

4.3. Validación del modelo

Antes de pasar a desarrollar el demodulador en VHDL es necesario comprobar si el modelo elegido cumple con las expectativas trazadas. Para ello se ha programado en Matlab un sistema que realiza la modulación y posteriormente la demodulación de una señal sinusoidal. A continuación se describe el programa de prueba.

4.3.1. Simulación con Matlab

Las componentes I y Q son generados como dos tonos, con una diferencia de fase de $\pi/2 \text{ rad}$ entre ellos, a una frecuencia f_0 .

```
%Señales I (Inphase) y Q (Quadrature)
I = cos(2*pi*fo*t);
Q = sin(2*pi*fo*t);
```

Se genera la señal recibida a frecuencia intermedia como:

```
%Señal IQ
IQ = I.*cos(w_fi*t) - Q.*sin(w_fi*t);
```

Que será la señal de entrada al demodulador IQ

```
phase = 2*pi*f_local*t+ph_local;
I = IQ.*cos(phase);    I = 2*filter(B,A,I); % Rama I
Q = -IQ.*sin(phase);  Q = 2*filter(B,A,Q); % Rama Q
```

Finalmente se realiza el mezclado y posterior filtrado de dicha señal. El filtro utilizado es un filtro FIR paso bajo por su estabilidad y nula distorsión de fase. El resultado de la simulación para una $f_0 = 10\text{MHz}$, una frecuencia de muestreo $f_s = 200\text{MHz}$, una frecuencia intermedia $f_{fi} = 30\text{MHz}$ y un filtro paso bajo de orden 32 (32 coeficientes) se presenta en la figura 4.3.

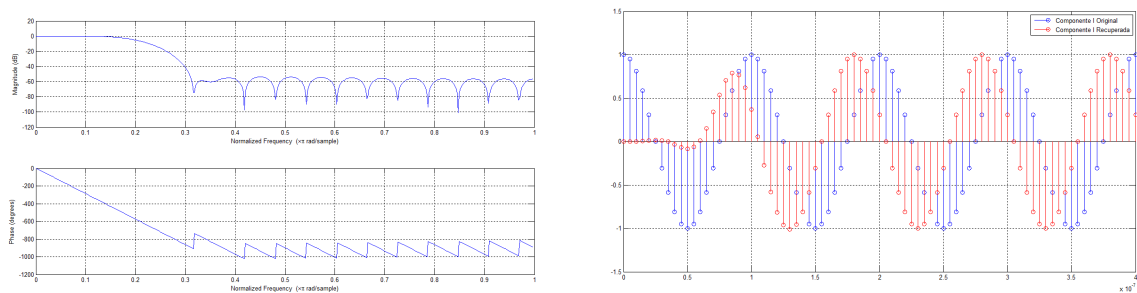


Figura 4.3: Módulo y fase del filtro y de la componente I.

De los resultados obtenidos en las diferentes simulaciones de este modelo se pueden sacar varias conclusiones que habrá que tener en cuenta en la siguiente fase:

- El ancho de banda de la señal debe ser mucho menor que la f_{FI} .
- La amplitud de la señal recuperada es $1/2$ de la señal original.
- La frecuencia de muestreo $f_s > 2BW$ (BW ancho de banda de la señal).
- El incremento del número de coeficientes del filtro hace su caída más abrupta en la zona de transición pero incrementa el retardo.
- La precisión de los coeficientes del filtro afectará su comportamiento y consumirá recursos de memoria.

En este punto cabe hacerse la pregunta si los recursos disponibles son suficientes para implementar dicha solución. Se procede a responder a esta cuestión desarrollando cada uno de los circuitos en un entorno de simulación como *Modelsim*.

4.3.2. Utilizando IP Cores

El demodulador Digital que se pretende diseñar consta de tres circuitos principales, el primero de ellos es un oscilador que genera las muestras de un tono a una frecuencia determinada, el segundo elemento es un mezclador cuya función es mezclar (multiplicar) la señal de entrada a frecuencia intermedia con el tono proporcionado por el oscilador y por último, un filtro paso bajo digital que se encarga de eliminar las componentes de mayor frecuencia.

En el diseño de estos circuitos se ha utilizado el generador de *IP Cores* de Xilinx porque permite optimizar los recursos de la FPGA y liberar el tiempo empleado en el desarrollo de funciones estándar.

Oscilador controlado numéricamente: También llamado sintetizador digital directo (DDS) incluye una tabla de búsqueda que contiene las muestras de una senoide, utiliza un integrador digital para generar un argumento de fase adecuado el cual es asignado por la tabla a la forma de onda de salida. Por medio de la interfaz de usuario se pueden configurar algunos parámetros como el número de bits de los datos de salida, su frecuencia y el nivel de supresión de espurios. En la figura 4.4 vemos un resumen de la configuración de este módulo, las simulaciones sobre el funcionamiento de este y de los siguientes circuitos se mostrarán en el capítulo 5. Se puede encontrar información mas detallada acerca de este núcleo en sus hojas de características [17].

Multiplicador: Configurado en modo *Paralelo* este circuito tiene como entradas las muestras de la señal de información y las muestras del DDS. A su salida presenta el producto de estas

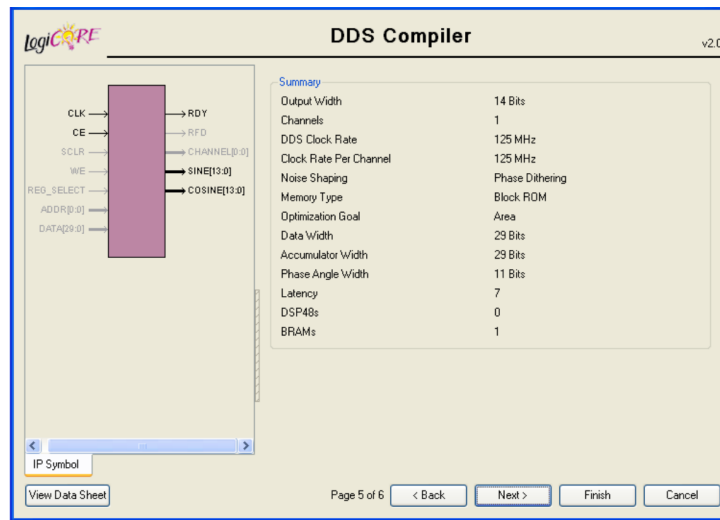


Figura 4.4: Configuración del NCO.

dos señales. Entre sus principales características se encuentra la aceptación de datos con signo, soporta anchos de palabra de 1 a 64 bits para la entrada y de 1 a 128 bits para la salida, además de un retardo configurable [19]. En la figura 4.5 se puede ver una representación del aspecto y configuración de este núcleo.

Filtro digital: Un filtro FIR paso bajo es la configuración mas básica que permite el núcleo utilizado. La figura 4.6 muestra el diagrama de bloque de una instancia de este módulo. Los datos de entrada del filtro se suministran por el puerto DIN y las muestras correspondientes a la salida se presentan por el puerto DOUT. Las señales ND, RFD y RDY son señales de control que permiten una interfaz al estilo flujo de datos simple y eficiente entre los datos de entrada y las muestras a la salida del filtro [18].

Los recursos utilizados por cada uno de los módulos en su configuración inicial y la integración de todos se muestran en la tabla 4.1 donde se observa que es el sintetizador digital quien más recursos demanda. Aunque el diseño se realizó pensando en optimizar dichos recursos se puede correr el riesgo de llegar al límite de utilización ya que cualquier modificación en la configuración de alguno de los circuitos, como puede ser un aumento en el número de coeficientes del filtro ó aumento del ancho de palabra de los puertos de salida, aumentaría los recursos utilizados sin que sea posible una ampliación futura del sistema. Esto se debe a que tanto el transmisor como el receptor se tendrán que implementar sobre la misma FPGA. Por tanto, es necesario buscar una solución más simple que reduzca de forma sustancial los medios requeridos.

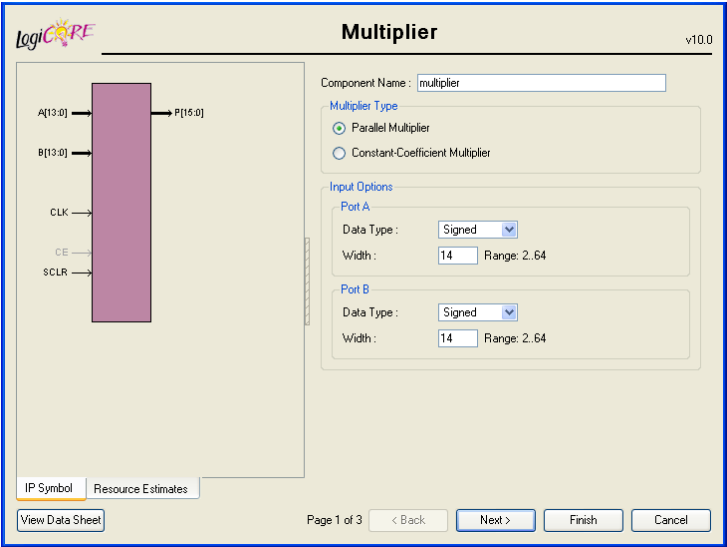


Figura 4.5: Configuración del multiplicador.

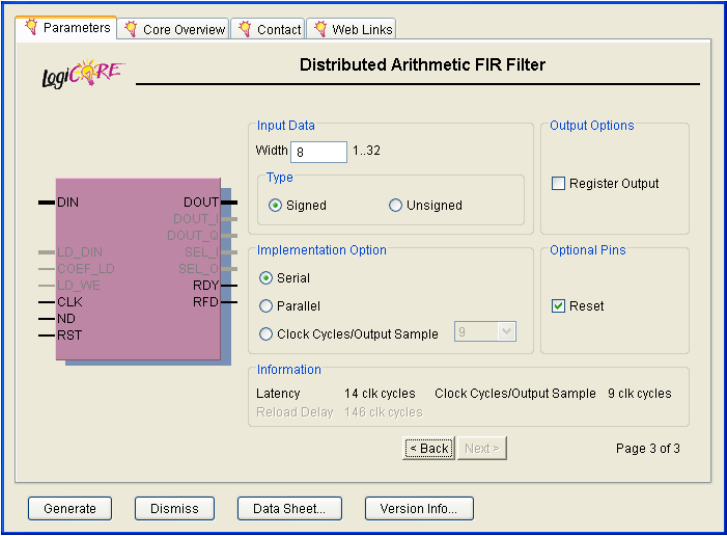


Figura 4.6: Esquemático de una instancia del filtro FIR.

Tabla 4.1: Resumen recursos utilizados.

	NCO	Multiplicador	Filtro	Integrado
<i>Flip-Flops</i>	200	30	100	480
<i>LUTs de 4 entradas</i>	142	218	59	3095
<i>FIFOs16/RAMs16</i>	1	0	0	2

4.4. Optimización de recursos

En esta sección se describirán los pasos que han permitido conseguir el objetivo trazado. Esto es, diseñar un demodulador Digital en cuadratura sencillo que haga un uso racional de los recursos disponibles adaptándose a las características de la señal y de la herramienta utilizada.

Si el circuito que mayor recursos demanda es el NCO, sería un buen comienzo replantear su diseño de forma que se reduzca su estructura. Si se representa un ciclo de una senoide mediante cuatro puntos (0, +1, 0, -1), es decir, cuatro muestras por periodo como se explica en [12], la salida digital del oscilador local sería una ristra de este tipo de valores tanto para el seno como para el coseno. Con esto se conseguiría un ahorro importante de recursos.

Pero se puede ir mas allá aprovechando el hecho de que todo el proceso de demodulación es digital. En [1] se hace un análisis de las muestras de la señal a frecuencia intermedia desde dos puntos de vista, uno teórico y otro de implementación. A continuación se hace un resumen de este estudio.

4.4.1. Análisis teórico

Como se explicó anteriormente, la señal a frecuencia intermedia es muestreada por un ADC, con n_b bits de cuantificación, a una frecuencia de muestreo f_s obteniendo a su salida una señal como la de la ecuación 4.2.

$$x_{IF}[n] = A[n] \cos(\Omega_{IF}nT_s + \Phi[n]) + N[n] \quad (4.2)$$

Donde :

A Envolvente de la señal.

T_s Periodo de muestreo.

Φ Fase de la señal.

N Ruido blanco Gaussiano.

La idea se basa en submuestrear la señal IF para obtener una réplica de la señal en banda base. Esto se puede llevar a cabo eligiendo f_{FI} y f_s adecuadamente.

$$\begin{aligned} f_{IF} &= k f_s \pm f_s/4, \quad \forall k \in \mathbb{Z}/k \geq 1 \\ f_s &\geq 4B \end{aligned} \tag{4.3}$$

Las relaciones arriba descritas establecen que no habrá solape en $f_s/4$. Por lo que una de las réplicas en $\pm f_s/4$ se puede llevar a banda base utilizando la propiedad de desplazamiento en frecuencia de la transformada de Fourier de una señal.

$$\begin{aligned} Z(f \pm f_0) &\xrightarrow{TF} Z[n] \cdot e^{\pm j2\pi f_0 n T_s} \\ f_0 &= f_s/4 \end{aligned} \tag{4.4}$$

Es decir, habrá que multiplicar la señal IF por $e^{-j\frac{\pi}{2}n}$. Esta operación es trivial ya que la secuencia de un ciclo de este factor es $\{1, -j, -1, j\}$, con lo que obtendremos las componentes I y Q de la señal en cuestión como se muestra a continuación:

$$\begin{aligned} x_{IF}[n] \cdot e^{-j\frac{\pi}{2}n} &= [x_{IF}[0], -jx_{IF}[1], -x_{IF}[2], jx_{IF}[3], x_{IF}[4], \dots] \\ x_Q[n] &= [x_{IF}[0], 0, -x_{IF}[2], 0, x_{IF}[4], 0, -x_{IF}[6], 0, \dots] \\ x_I[n] &= [0, -x_{IF}[1], 0, x_{IF}[3], 0, -x_{IF}[5], 0, x_{IF}[7], \dots] \end{aligned}$$

4.4.2. Diseño para FPGA

Si se elige adecuadamente la frecuencia de muestreo, cumpliendo las relaciones de la ecuación 4.3, implementar un demodulador de estas características es una operación muy sencilla en VHDL. Este proceso sólo requiere separar las muestras pares e impares y luego multiplicar cada salida por una secuencia cíclica de valores ± 1 .

La implementación del demodulador se desarrolla en dos fases, la primera se basa en un demultiplexor que obtiene las muestras pares por una de sus salidas y las muestras impares por otra, la segunda se encarga de asignar un signo a cada muestra de salida. Notar que esta última operación se puede realizar a $f_s/2$ ya que los valores nulos entre medias (a la salida del demultiplexor) no necesitan ser multiplicados.

En la figura 4.7 se muestra el diagrama de bloques del demodulador IQ implementado. Con esta solución queda asegurada la optimización de recursos pues solo se hace uso de un demultiplexor 1:2 y la asignación de signos se hace a través de un complemento a dos para aquellas

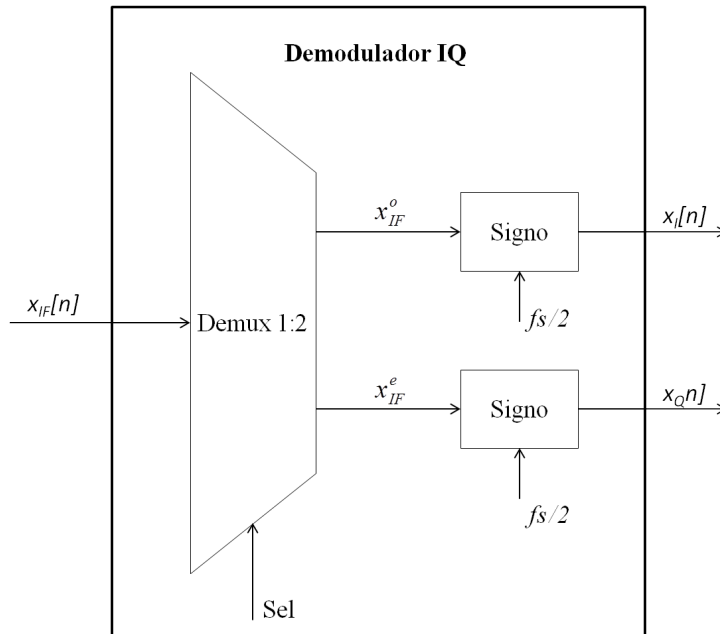
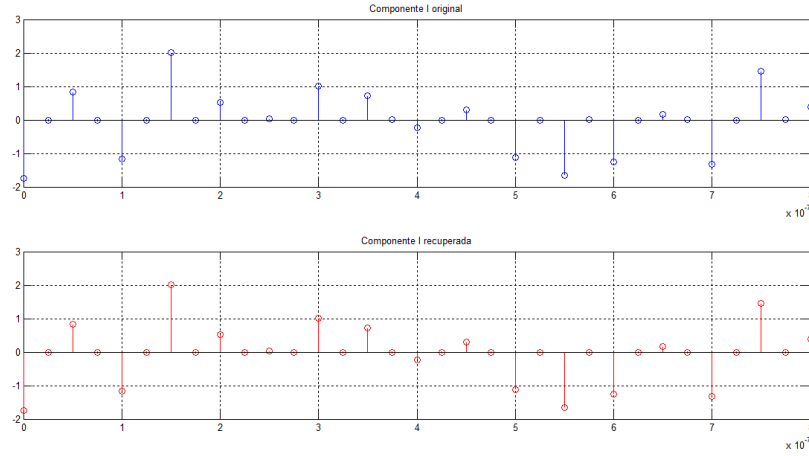
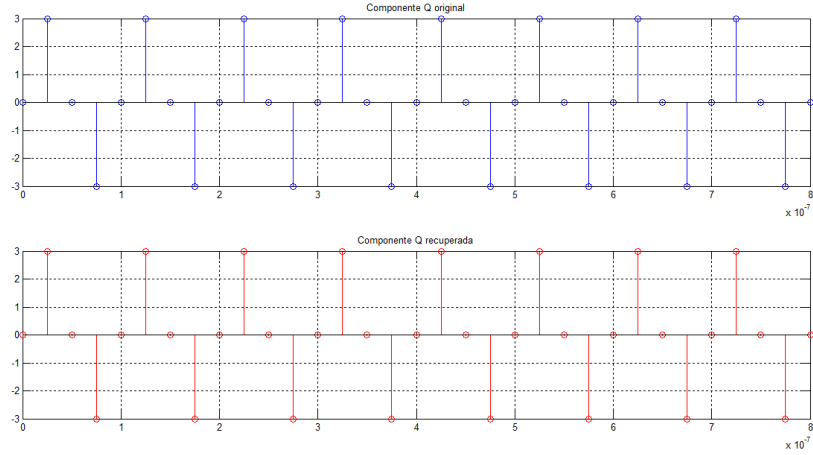


Figura 4.7: Diagrama de bloques del demodulador IQ implementado.

muestras que deban ser multiplicadas por -1. Con lo que se evita la utilización de multiplicadores que consuman recursos.

Nuevamente, se valida el funcionamiento teórico mediante *Matlab*, para ello se genera un fichero de prueba como el que se diseñó para la primera versión del demodulador. Los resultados que se muestran en las figuras 4.8 y 4.9 se obtuvieron para una configuración como la siguiente:

```
%frecuencia de muestreo
fs = 40e6;
%frecuencia de señal
fo = 10e6;
%linea temporal para mostrar 8 periodos de señal
t = [0:1/fs:8*(1/fo)];
%Señales I (Inphase) y Q (Quadrature)
I = randn(1,length(t)).*cos(2*pi*fo*t);
Q = 3*sin(2*pi*fo*t);
%Frecuencia intermedia
f-fi = 30e6;
w-fi = 2*pi*f-fi;
```

Figura 4.8: *Recuperación de la componente I.*Figura 4.9: *Recuperación de la componente Q.*

%Señal IQ

```
IQ = I.*cos(w_fi*t) - Q.*sin(w_fi*t);
```

Como se puede ver, la frecuencia de muestreo se eligió tal que fuese $\frac{4}{3}f_{fi}$ y la relación entre esta y la frecuencia de la señal fuese de un factor 4, por lo que se cumple las condiciones impuestas en 4.3, con $k = 1$. Las amplitudes de las componentes I y Q son diferentes, la primera se probó con una amplitud aleatoria y la segunda con una amplitud constante para observar el comportamiento en cada caso. Los resultados que arroja esta prueba son contundentes y no deja duda que esta es la mejor solución a implementar ya que consume el mínimo de recursos como se puede ver en la tabla 4.2, no hay distorsión ni retraso en las señales recuperadas.

Tabla 4.2: *Resumen recursos utilizados por el demodulador implementado.*

	Demodulador Digital I-Q
<i>Flip-Flops</i>	61
<i>LUTs de 4 entradas</i>	87
<i>Número de puertas equivalentes para el diseño</i>	1378

Pruebas y Resultados

En este capítulo se describirán las pruebas y mediciones realizadas para comprobar el funcionamiento del sistema así como de cada uno de los módulos sobre los que se actuaron. Se empezará mostrando los resultados obtenidos para una transmisión como la que se describe en la anterior fase pero que recoge las modificaciones realizadas en esta. A continuación, se mostrarán los resultados alcanzados con la configuración del módulo de radiofrecuencia, seguidamente se analizarán las pruebas de amplificación de la señal transmitida y recibida. Finalmente, se expondrán las pruebas del modulador digital en cuadratura.

5.1. Transmisión OFDM

En esta sección se mostrarán los resultados obtenidos en la transmisión y recepción de un conjunto de tramas que contienen, cada una, cien símbolos OFDM ($N = 64$), un prefijo cíclico de longitud ocho y una secuencia corta de entrenamiento de 160 símbolos como se explicó en el Capítulo 3. El reloj del transmisor trabaja a 9,375 MHz y cada símbolo está amplificado por un factor 2×10^3 .

Se empezará con la señal en banda base, luego se mostrarán los resultados en radiofrecuencia y se acabará con las imágenes de la señal recibida a frecuencia intermedia.

5.1.1. Medidas en Banda Base

En las figuras 5.1 y 5.2 se muestra la secuencia corta de entrenamiento de una trama OFDM, en ellas podemos comprobar que la frecuencia de transmisión es de 9,375 MHz tal y como está



Figura 5.1: *Frecuencia del reloj del transmisor.*

configurado en el módulo del transmisor, este dato también corresponde al ancho de banda de la señal. Además, vemos como la duración de dicha secuencia es de $17,06 \mu s$ que corresponde a lo esperado si multiplicamos los 160 símbolos del STS con el periodo de reloj. La amplitud de este tramo de señal está en el rango de los $334 mv$ y es menor que el resto de la trama.

En la figura 5.3 podemos ver una trama completa como se describió anteriormente. Se puede apreciar que la duración de esta es aproximadamente $786 \mu s$ resultado muy próximo al teórico que resulta de multiplicar el número de símbolos de la trama por el periodo de reloj con las que estos fueron generados. Es decir:

$$(100 * 64 + 100 * 8 + 160) / 9,375 \times 10^6 Hz = 785,06 \mu s$$

En la figura 5.4 se mide el tiempo que tarda el modulador en procesar una trama, esto es, el tiempo que toma convertir 128 bits en 64 símbolos QPSK y obtener con estos un símbolo OFDM al que se le suma un prefijo de longitud 8, repetir este proceso hasta conseguir 100 símbolos OFDM a los que se les suma una secuencia corta de entrenamiento. Además, se puede observar que la amplitud de las tramas es de aproximadamente $780 mv_{pp}$.

Pasamos al dominio de la frecuencia, en la figura 5.5 se puede constatar que el ancho de banda de la señal en banda base es de 9.375 MHz entre nulos y que el segundo armónico está

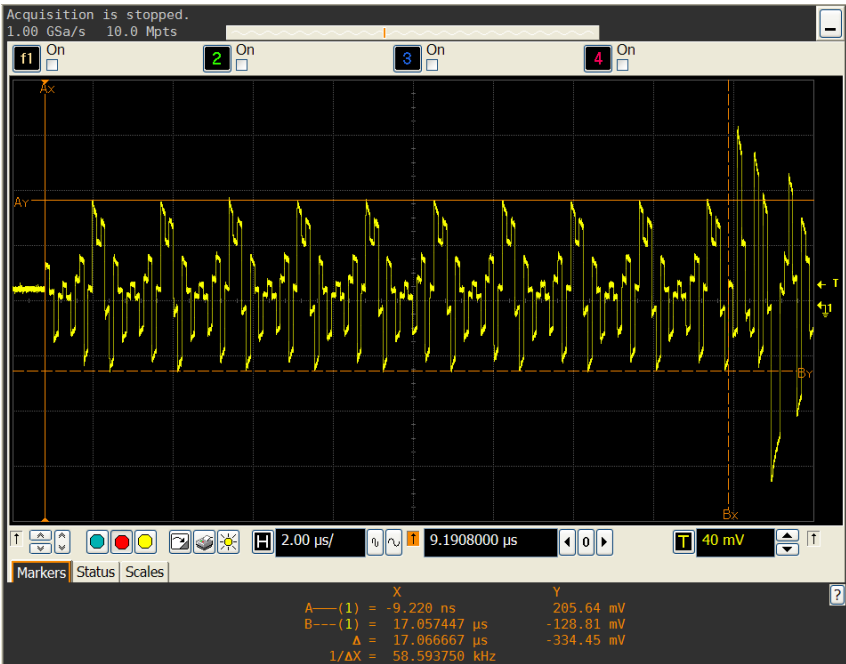


Figura 5.2: *Secuencia corta de entrenamiento.*

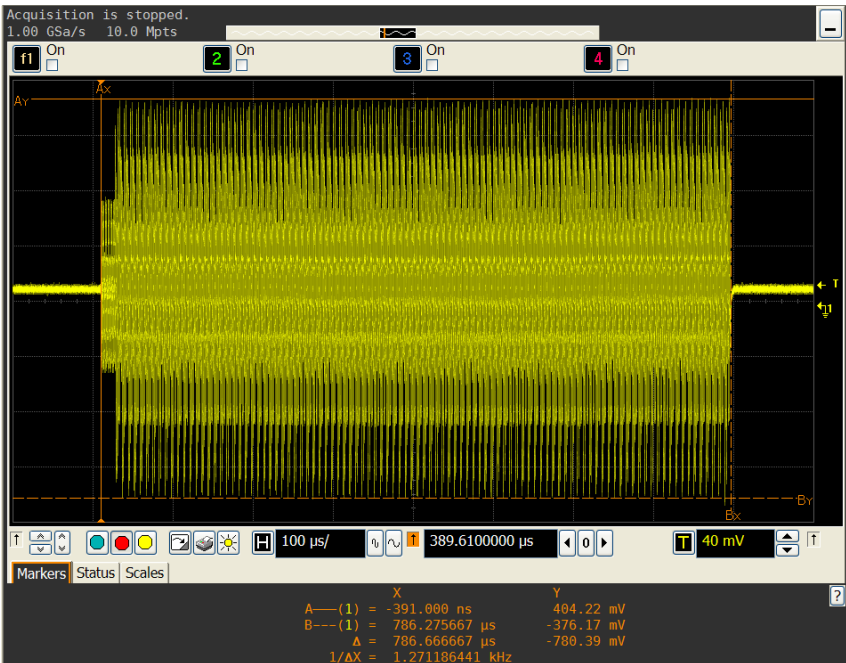


Figura 5.3: *Trama de 100 símbolos OFDM.*

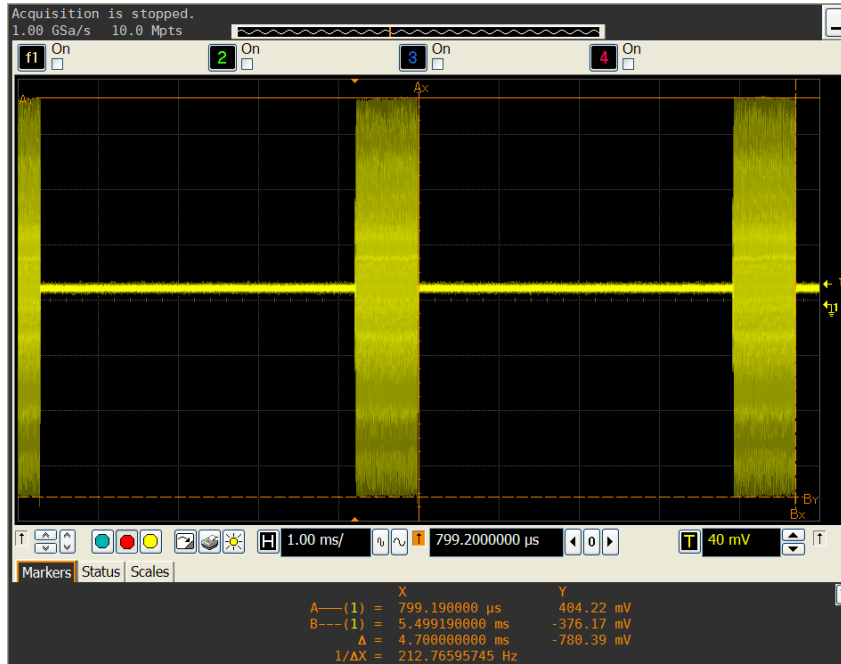


Figura 5.4: *Tiempo de procesamiento del modulador.*

11.2 dB por debajo del armónico principal.

5.1.2. Señal radio

En la figura 5.6 se puede observar la señal transmitida en radiofrecuencia, esto es la suma de la parte real e imaginaria de los símbolos OFDM sobre una portadora de 370 MHz, podemos comprobar que la estructura se mantiene, el tiempo entre tramas es idéntico al anterior y la amplitud es de 948 mv_{pp} aproximadamente la cual ha aumentado gracias a la contribución de los dos canales.

El espectro de la señal RF se puede ver en la figura 5.7, en ella se aprecia el primer armónico a 370 MHz mientras que el segundo armónico se sitúa a 470 MHz con un nivel de 12 dB inferior al principal. La figura 5.8 muestra una ampliación del espectro que permite apreciar el ancho de banda que en efecto es el doble del que presenta en banda base.

5.1.3. Señal recibida a FI

En esta fase nos interesa conocer las características de la señal recibida para deducir los efectos del canal, las amplificaciones y atenuaciones de los distintos módulos por los que ha pasado la señal.

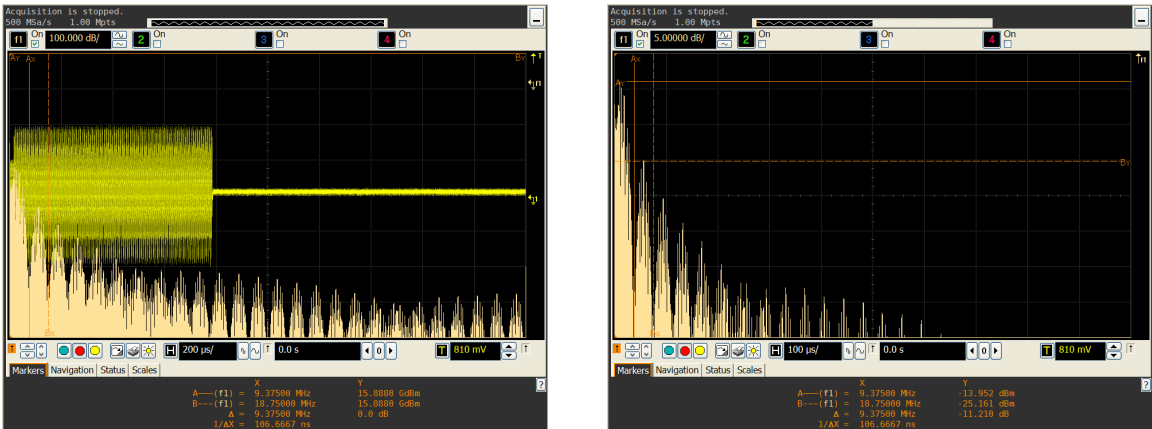


Figura 5.5: Espectro de la señal Banda-Base.

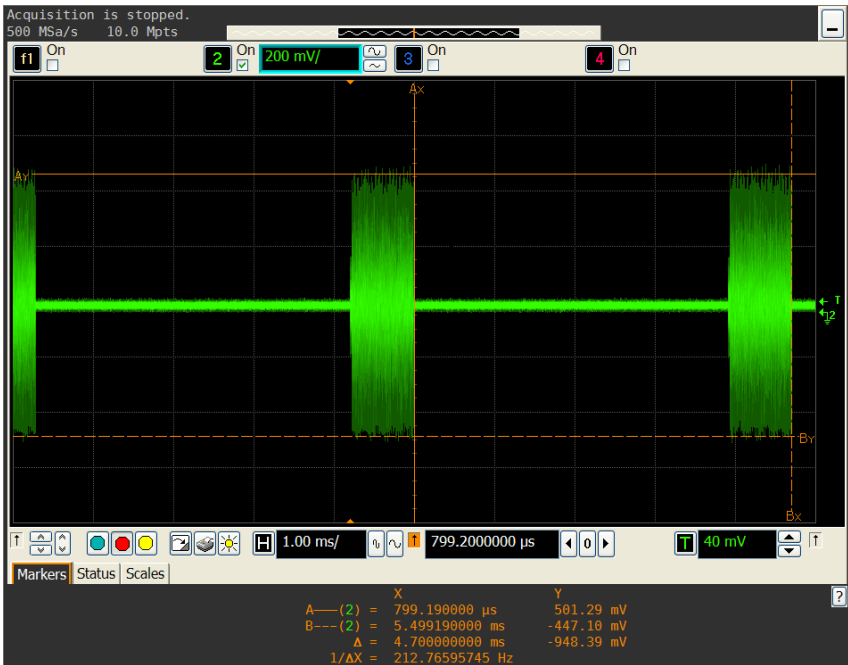


Figura 5.6: Señal RF.

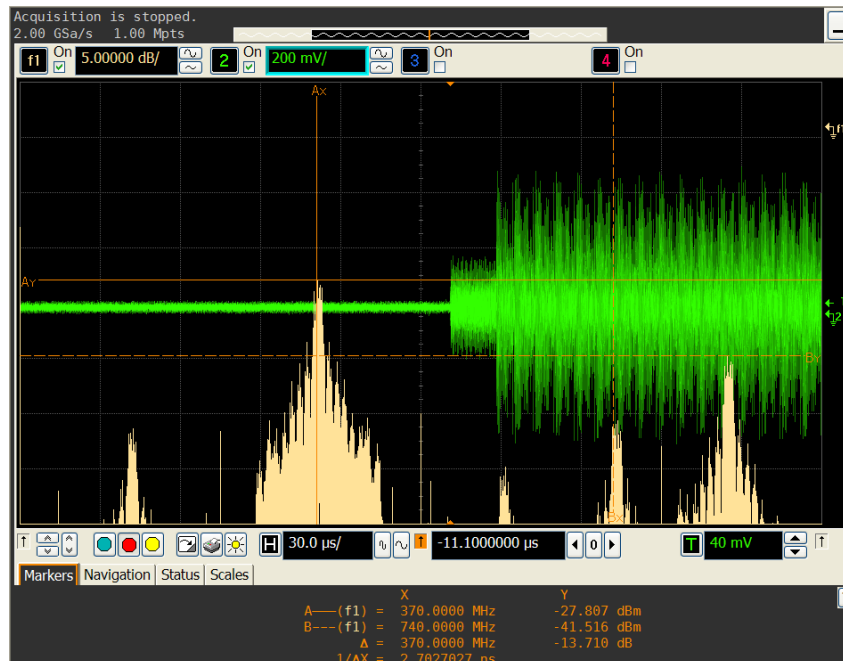


Figura 5.7: Espectro de la señal transmitida.

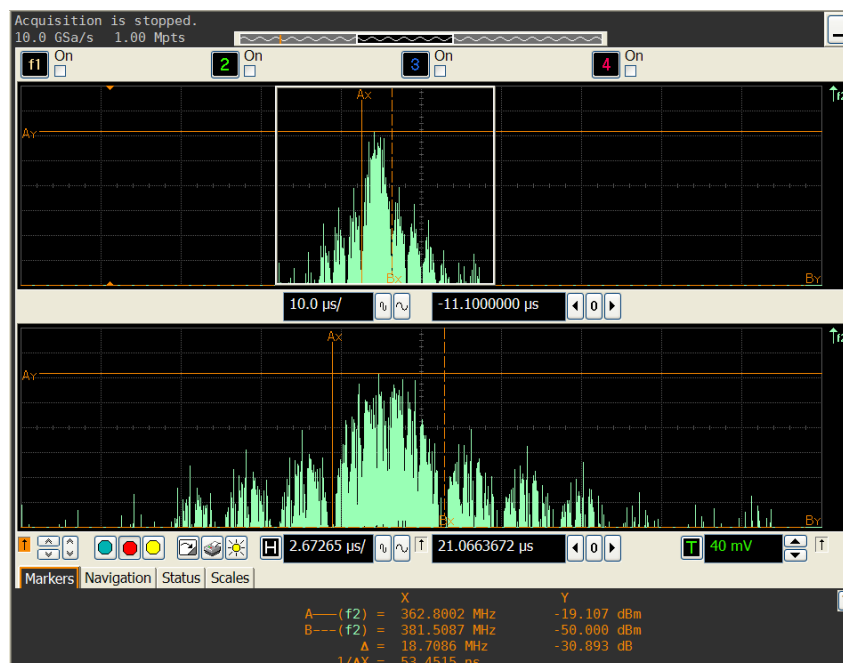


Figura 5.8: Espectro de la señal transmitida (Ampliación).

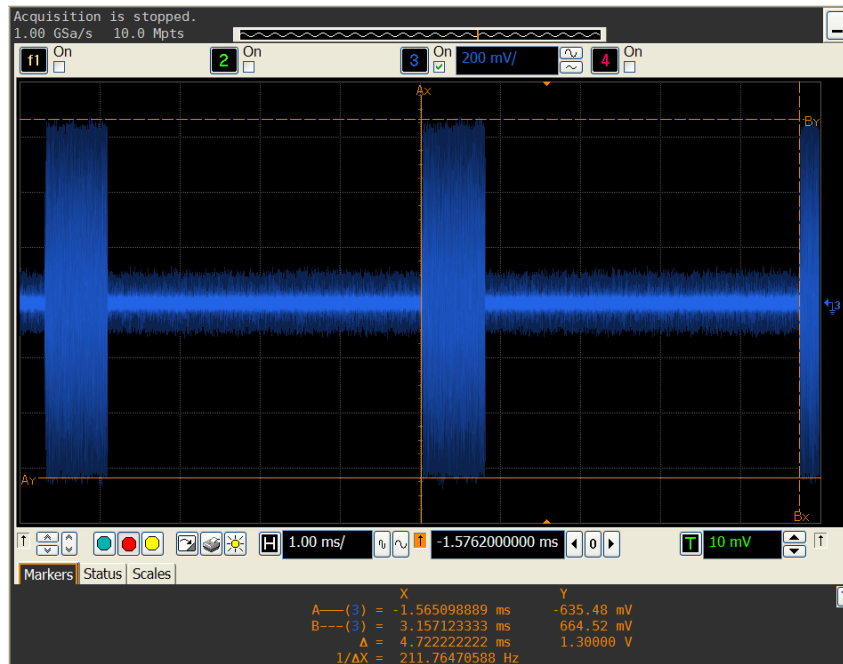


Figura 5.9: Señal a frecuencia intermedia.

En la figura 5.9 se puede observar que la amplitud de la señal recibida ha aumentado respecto a la señal RF unos 350 *mv*, esta ganancia la proporcionan las antenas tanto transmisora como receptora, cada una de ellas aporta alrededor de 2.5dBi de ganancia.

La figura 5.10 se obtuvo al cubrir con la mano la antena transmisora para ver como se atenúa la señal recibida en este caso. Las pérdidas pueden llegar a ser más del 50 % del valor original.

El espectro de la señal IF se puede ver en la figura 5.11 y 5.12, en la primera el marcador *Ax* indica la frecuencia de la señal, en este caso como se explicó en la sección herramientas Hardware, el módulo receptor RF se configura para que la señal sea convertida a una frecuencia de 30MHz. También podemos ver la diferencia de 16 dB entre el armónico principal y secundario. En la segunda figura se ha mejorado la resolución para ver el ancho de banda de la señal.

5.2. Funcionalidad de la herramienta

En esta sección se presentan los resultados obtenidos al realizar diversas configuraciones sobre el dispositivo SDR. Se mostrarán los límites de funcionamiento relacionados con el máximo ancho de banda en transmisión así como las bandas de frecuencia en las que se puede transmitir y se presentarán las medidas realizadas con diferentes niveles de ganancias programadas en el módulo

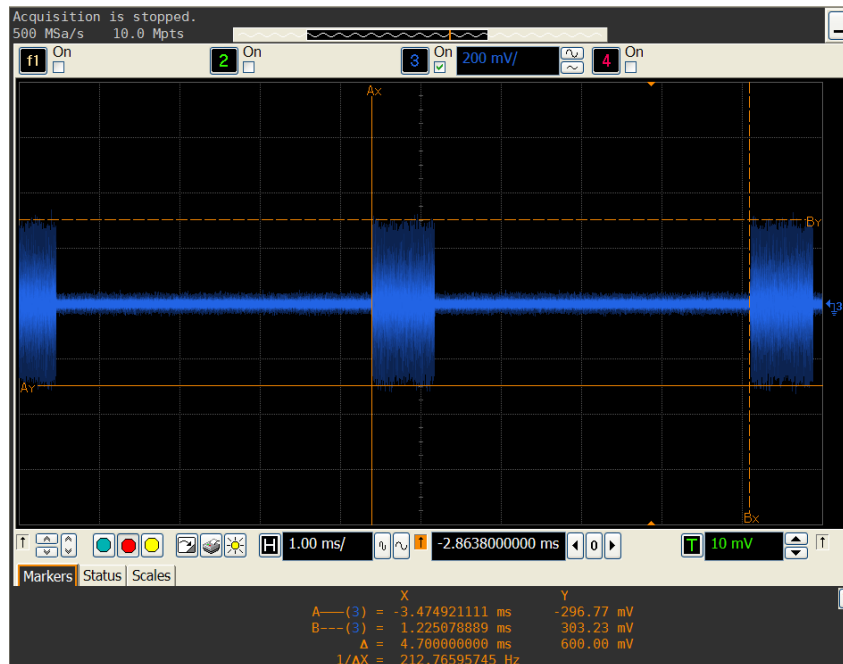


Figura 5.10: Atenuación de la señal recibida.

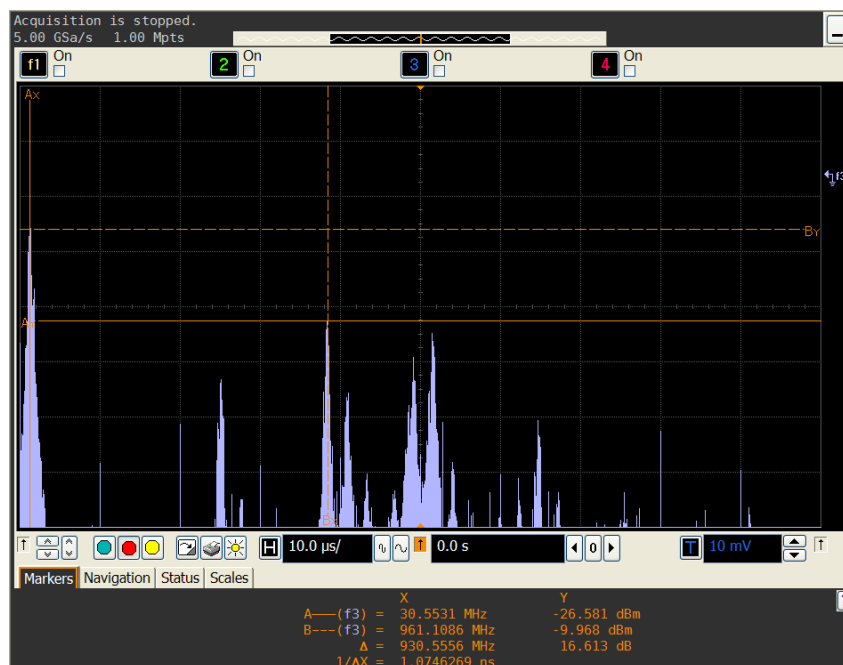


Figura 5.11: Espectro de la señal recibida.

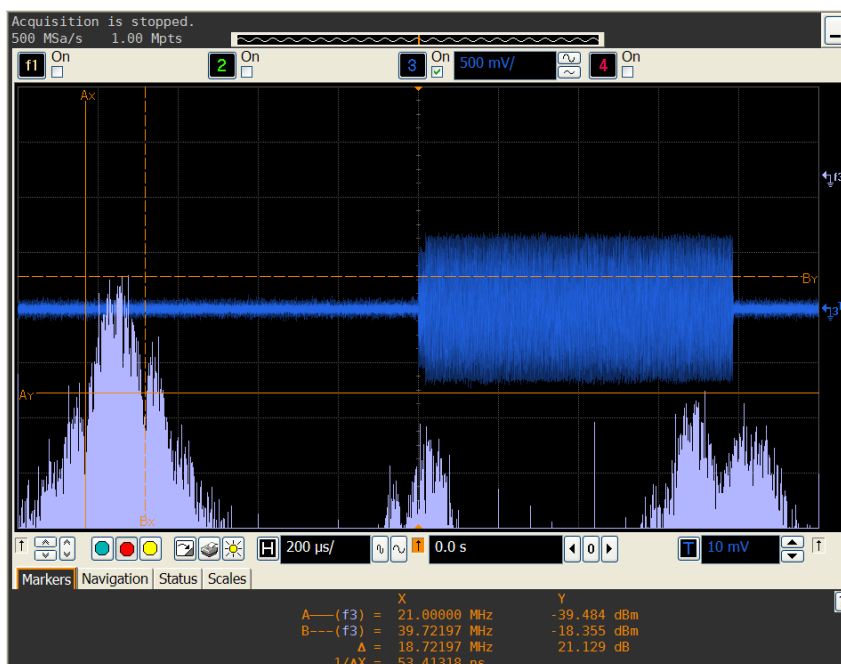


Figura 5.12: Ampliación del espectro de la señal recibida.

de conversión de datos.

Para comprobar cuáles son los límites de funcionamiento de nuestro sistema se ha realizado una prueba que consiste en transmitir a una frecuencia mayor de 60 MHz, como se vio en el Capítulo 2, el módulo de radiofrecuencia en sentido transmisión presenta un filtro paso bajo que limita la banda de la señal transmitida. En la figura 5.13 se observa que cuando se transmite a una frecuencia superior al ancho de banda del filtro, la señal enviada sufre una distorsión muy importante.

La siguiente prueba consiste en constatar la posibilidad de transmitir en las bandas GMRS y FRS. Para ver que es posible transmitir a una frecuencia comprendida en la primera banda (262-438 MHz) se puede recurrir a la figura 5.7 donde se transmite a una frecuencia de 370 MHz. La figura 5.14 muestra la transmisión a 740 MHz como lo indica el marcador Bx, esta frecuencia se encuentra dentro de la segunda banda (523-876 MHz).

Pero ¿Qué ocurre si se transmite a una frecuencia que se encuentra fuera de estas dos bandas? En la figura 5.15 podemos ver lo que ocurre cuando se transmite a una frecuencia comprendida entre el límite superior de la primera banda y el límite inferior de la segunda, en este caso 480 MHz, si se sitúa el cursor en esta frecuencia vemos como el espectro no está centrado en ella, es decir, tenemos una deriva en la frecuencia que proporciona el oscilador local y por tanto esta

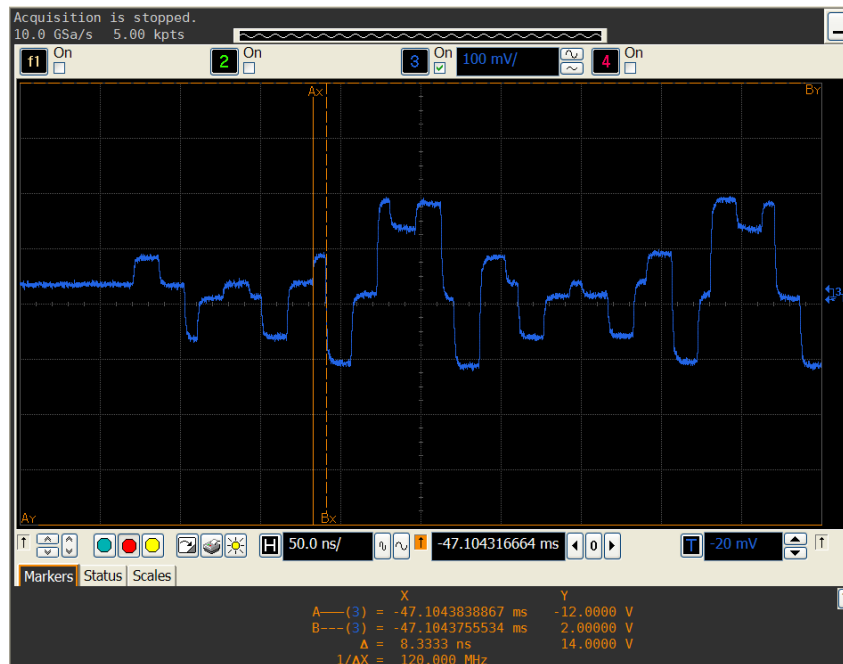


Figura 5.13: Frecuencia de transmisión mayor de 60 MHz.

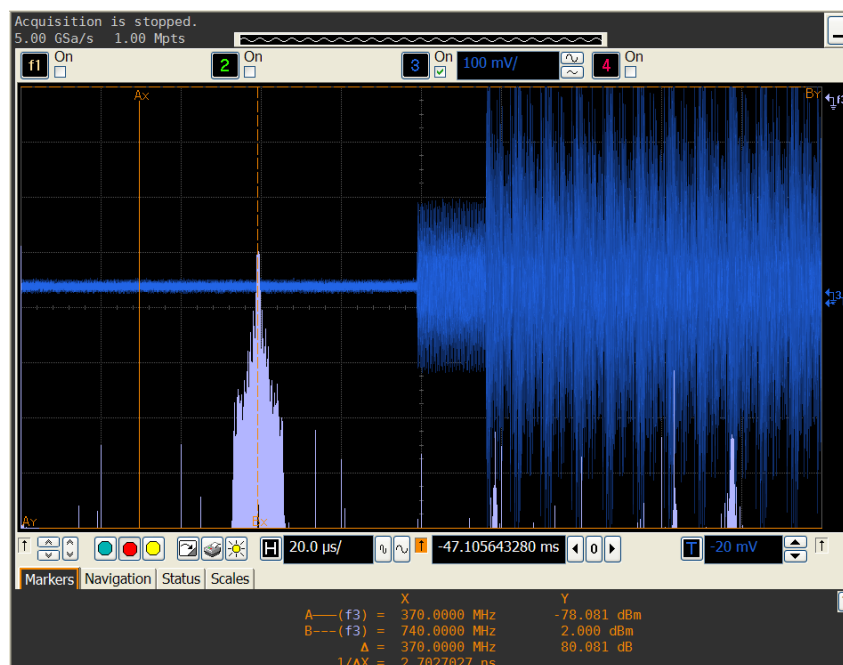
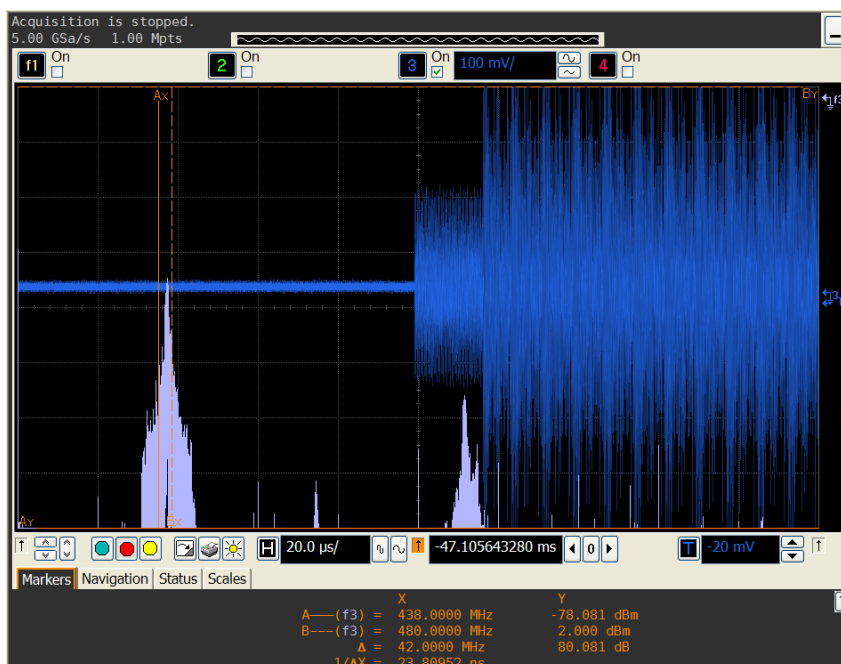


Figura 5.14: Transmisión a 740 MHz.

Figura 5.15: *Transmisión a 480 MHz.*

señal no es estable.

Sí se realiza la misma prueba para una frecuencia que está por encima del límite superior de la segunda banda vemos que el resultado es muy similar, en este caso se aprecia la existencia de inestabilidad a partir de 900 MHz como se puede ver en la figura 5.16.

Por tanto al realizar estas pruebas queda constatado que el ancho de banda máximo en transmisión es de 60 MHz y que solo se puede transmitir en el rango de frecuencias de las bandas GMRS y FRS.

A continuación, en la tabla 5.1 se registran los valores de configuración de la ganancia del DAC y posteriormente se procede a comentar los resultados obtenidos.

Con la primera configuración obtenemos una amplitud en el canal B de 57 mv_{pp} y se observa que en recepción no está saturado el ADC, al pasar a la segunda configuración la amplitud de este canal pasa a ser de 62 mv_{pp} , es decir, se consigue un aumento de 5 mv_{pp} , si además se aumenta al máximo el ajuste fino como se ve en la tercera configuración, se consigue aumentar la señal aproximadamente $2,5\text{ mv}_{pp}$. Esto es, al incrementar el ajuste grueso en un punto y poner al máximo el ajuste fino se consigue un incremento del 13 % de la amplitud de la señal.

Con la segunda configuración se midió la amplitud de la señal RF (combinación de la componentes I y Q) en el conector de la antena transmisora siendo esta de $65,5\text{ mv}_{pp}$ y la señal recibida

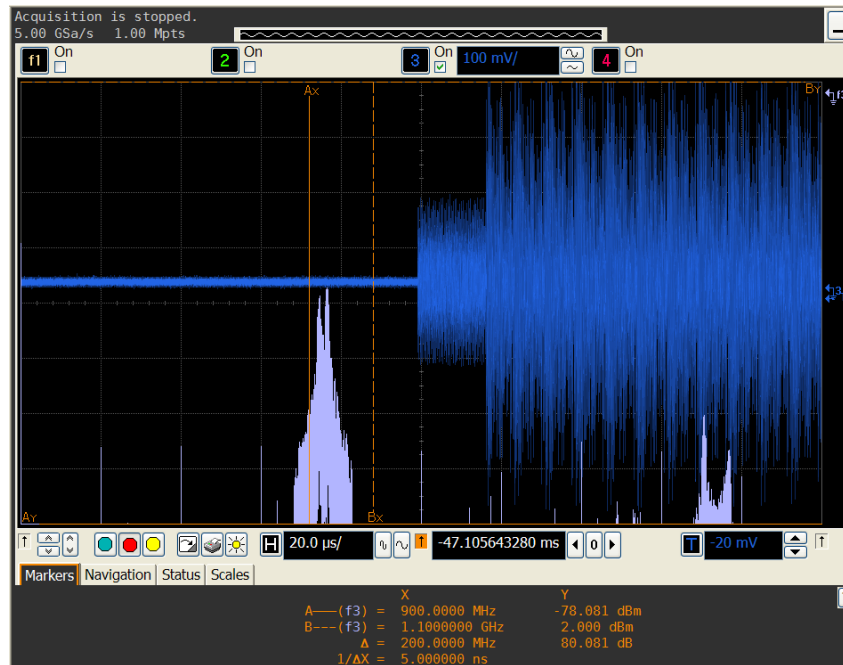


Figura 5.16: Transmisión a 1,1 GHz.

Tabla 5.1: Amplificación DAC

i	Canal A		Canal B	
	COARSE	FINE	COARSE	FINE
1	14	0	14	0
2	15	0	15	0
3	15	127	15	127
4	0	0	0	0
5	4	0	4	0

(medida justo después de la antena receptora) presentó una amplitud de 29 mv_{pp} , si suponemos que cada antena aporta una ganancia de 2.5 dBi podemos hacer una estimación de las pérdidas sufridas en el trayecto. Realizando el cálculo oportuno se obtiene que dicha estimación es 12 dB lo que podemos atribuir a las pérdidas del espacio libre y pérdidas por desadaptación y desacoplo.

Las dos siguientes configuraciones tienen el objetivo de determinar cuál es la mínima amplificación que permite diferenciar la señal del ruido, aumentando el valor del ajuste grueso en incrementos de un punto se obtiene que a partir de un valor 4 podemos observar esta diferencia, cabe destacar que la señal digital es preamplificada por un factor 100 antes de pasar por el DAC.

Por último, si se incrementa el factor del preamplificado en un orden de magnitud (1000) y configuramos al máximo los ajustes grueso y fino observamos saturación en el buffer de recepción.

5.3. Simulación del demodulador IQ

A continuación se presentarán los resultados obtenidos por simulación de las dos versiones del demodulador digital IQ desarrolladas. En la primera parte se presentan las pruebas realizadas a cada uno de los bloques diseñados a través del generador de *Cores* de Xilinx y posteriormente se muestran las pruebas realizadas sobre la solución que permite una mejor utilización de los recursos de la FPGA.

5.3.1. Vamos por partes

Para comprobar el funcionamiento del desarrollo inicial se ha procedido a probar cada uno de los módulos que lo componen. De esta forma, en la figura 5.17 se muestra una captura de pantalla de la simulación del NCO, en ella podemos ver que la frecuencia de trabajo es de 125 MHz , las salidas de datos son de 14 bits, la primera muestra válida se obtiene después de 9 ciclos de reloj indicado por la señal *rdy* activa a nivel alto y que hay cuatro muestras por periodo, lo que indica que la frecuencia del oscilador local es de $31,25\text{ MHz}$.

Para comprobar el correcto comportamiento del multiplicador se ha conectado a sus dos entradas una de las salidas del NCO, obteniendo como resultado el cuadrado del coseno. En la figura 5.18 se presenta la simulación de este bloque, en esta se puede ver que el retardo en la recepción de los datos se debe al NCO, una vez que se tienen datos en sus entradas se realiza la multiplicación y se tarda en mostrar el resultado un ciclo de reloj.

Una forma de comprobar el funcionamiento general del filtro es mediante la respuesta al

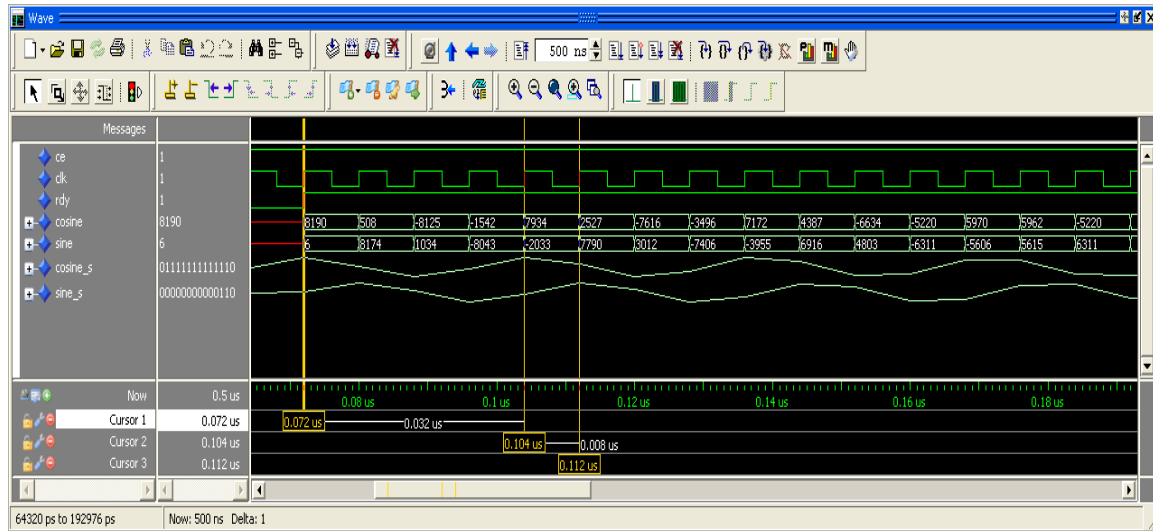


Figura 5.17: Simulación del NCO.

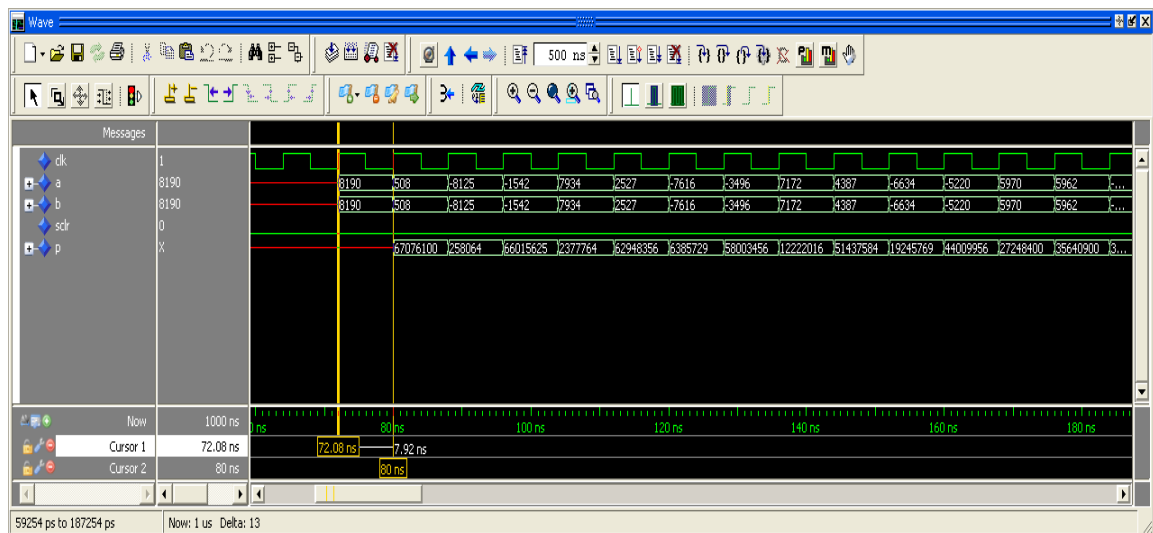
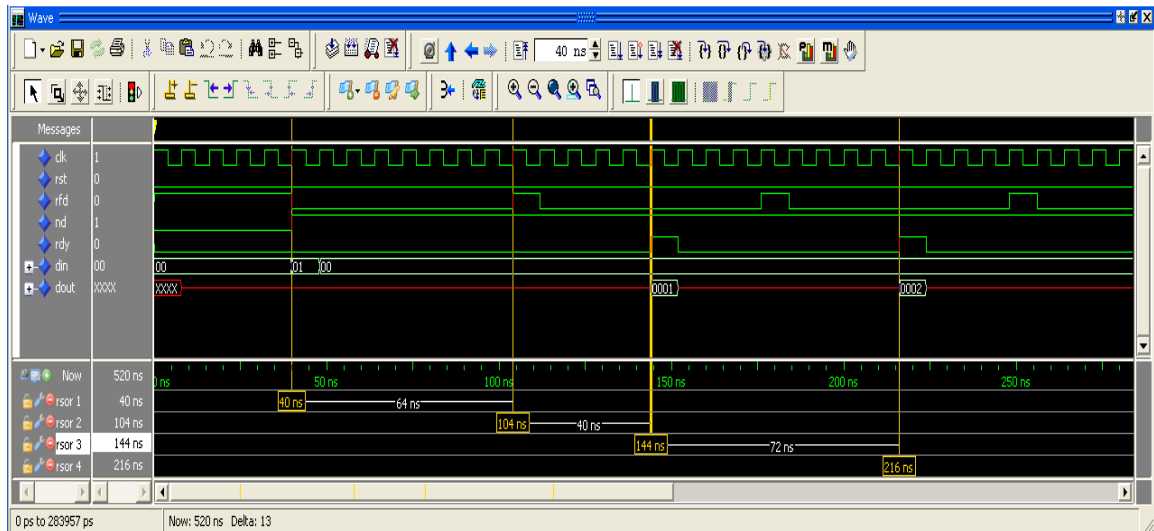


Figura 5.18: Simulación del Multiplicador.

Figura 5.19: *Simulación del Filtro.*

impulso. La prueba consiste en introducir un valor no nulo durante un ciclo de reloj rodeado por ceros en ambos lados, simulando así un pulso. En la figura 5.19 se puede ver los resultados obtenidos para esta simulación, como era de esperar, a la salida del filtro obtenemos el valor de sus coeficientes y esto sucede cuando la señal *rdy* está activa a nivel alto.

5.3.2. Optimizando recursos

En este apartado se presentan las pruebas realizadas sobre el demodulador que permite optimizar los recursos de la placa. En la figura 5.20 podemos observar la simulación que se hizo sobre este módulo. El banco de prueba es muy sencillo, puesto que el diseño también lo es. Este consiste en introducir una ristra de números los cuales serán separados por un demultiplexor en función de su posición (muestra par o impar). A las salidas *demiq_inphase* y *demiq_quadrature* podemos ver la correcta asignación de símbolos siguiendo el procedimiento explicado en la parte final del Capítulo 4. Cabe destacar que en esta prueba no se ha tenido en cuenta lo de hacer la asignación de símbolos a una frecuencia mitad de la del demultiplexor con el objetivo de ver con mas claridad que los resultados obtenidos son idénticos a los planteados teóricamente.

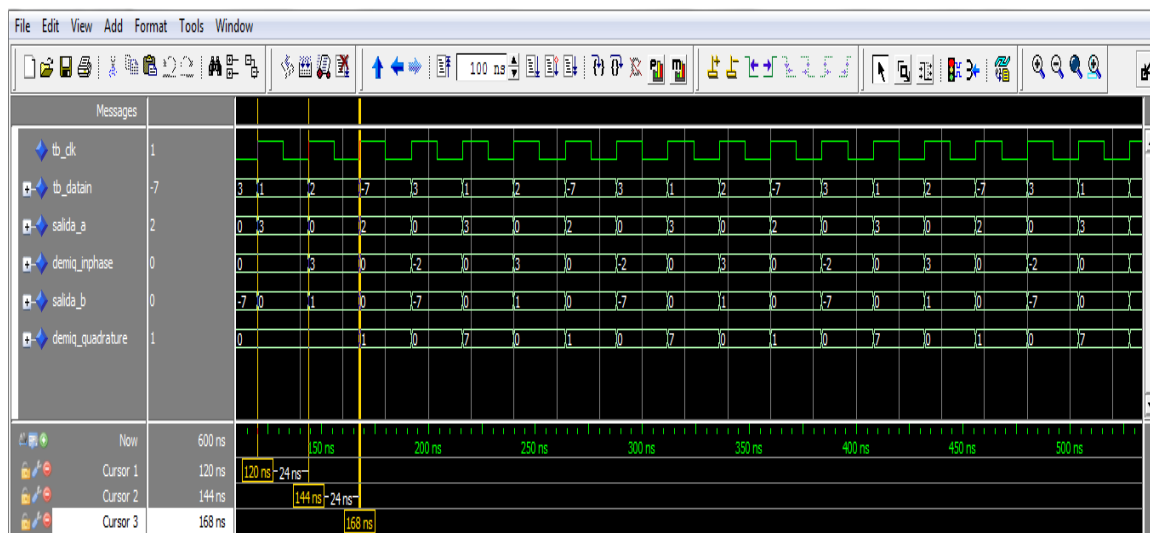


Figura 5.20: *Simulación del Demodulador Digital implementado.*

Capítulo 6

Conclusiones

En este capítulo se hará una descripción resumida y una valoración crítica del conjunto del trabajo realizado. Se detallarán las aportaciones de este proyecto al propósito global evaluando el cumplimiento de los objetivos. A continuación se plantearán las posibles ampliaciones y mejoras del trabajo desarrollado. Finalmente se presentarán a modo de anécdota las incidencias mas relevantes ocurridas durante este trabajo.

Este proyecto ha dado continuidad al desarrollo de un sistema de comunicaciones radio que utiliza una técnica de acceso al medio como OFDM, la programación se ha hecho sobre la FPGA de un dispositivo SDR para el cual ha sido necesaria la configuración de cada uno de sus tres principales módulos a saber, el módulo DSP para configurar y controlar todos los dispositivos de la placa, el módulo de adquisición de datos que configura el DAC y el ADC y el módulo de radiofrecuencia que permite la transmisión y recepción de las señales. Para hacer posible el ciclo de recepción se diseño un demodulador digital en cuadratura en un lenguaje de alto nivel como VHDL.

En el desarrollo de esta fase se ha logrado implantar un modelo de trabajo modular que permite el diseño de las diferentes partes que integren el sistema de forma independiente. Este proyecto no concluye el sistema de comunicación completo dejando los módulos y tratamientos faltantes para futuras ampliaciones o fases posteriores.

Para llevar a cabo esta tarea fue necesario el estudio de los conceptos relacionados con la transmisión de señales por canales radio, conocer las características técnicas del dispositivo SDR, aprender el funcionamiento de las plataformas software que se utilizaron para la configuración de la herramienta y la implementación del diseño. Además, diseñar un demodulador digital en

cuadratura que utiliza los mínimos recursos disponibles. Este demodulador permite recuperar las señales en fase y en cuadratura de la señal recibida y aplicar el proceso inverso al de transmisión.

Por todo esto se considera que los objetivos propuestos fueron alcanzados en su totalidad y para constancia de ello se realizaron diversas medidas en tiempo y frecuencia de la señal transmitida y varias pruebas y simulaciones sobre los módulos diseñados. Estas pruebas no sólo sirvieron para validar el correcto funcionamiento de los desarrollos sino que también permitió conocer las limitaciones físicas de la herramienta.

6.1. Líneas futuras

En este apartado se plantearán posibles mejoras y ampliaciones para las siguientes fases del proyecto:

- Diseñar un receptor que contenga al demodulador para que se comunique con el transmisor. Completar el sistema OFDM desarrollando los módulos de inclusión y extracción de pilotos, el controlador de ganancia automática, y el módulo de sincronización de trama y estimación/corrección del offset.
- Cambios en la configuración de los parámetros de la señal transmitida tales como el número de símbolos transmitidos por trama, número de portadoras (tamaño de la IFFT y la FFT), cambios en la constelación aumentando a 16 QAM ó 64 QAM.
- Una vez completado el sistema OFDM se podrá realizar estudios de calidad de la información evaluando la BER, la SNR y el control de ganancia. También se puede caracterizar el medio de transmisión.
- Sobre el dispositivo SDR se puede implementar cualquier otro estándar de comunicación inalámbrica como TETRA, APCO-25, WIMAX, UMTS, cdma2000, LTE, etc.

6.2. Dificultades

La curva de aprendizaje fue llana durante la primera mitad del desarrollo de esta fase debido a la escasa información disponible sobre la herramienta y la poca información técnica que ofrecen los manuales del dispositivo.

Gran parte del tiempo de este proyecto se ha dedicado a comprender el funcionamiento de los distintos módulos hardware y como interactúan estos entre sí para su posterior configuración. La mayor dificultad se ha encontrado en el manejo de la librerías que intervienen en el proceso de implementación de un módulo hardware sobre una FPGA y la integración de las distintas plataformas de desarrollo (ISE, CCS y Modelsim).

Finalmente se recomienda la adquisición de otros dispositivos con estas características para realizar trabajos en paralelo y así poder compartir las experiencias y el conocimiento dentro del grupo de trabajo.

APÉNDICES

Presupuesto Del Proyecto

En este apéndice se detallan los costes globales de la realización de este Proyecto Fin de Carrera. Tales costes, imputables a gastos de personal y de material, se pueden deducir de las Tablas A.1 y A.2.

En la Tabla A.1 se muestran las fases del proyecto y el tiempo aproximado para cada una de ellas. Así pues, se desprende que el tiempo total dedicado por el proyectando ha sido de 1.184 horas, de las cuáles aproximadamente un 10 % han sido compartidas con el tutor del proyecto, por lo que el total asciende a 1.302 horas. Teniendo en cuenta que el coste por hora de un Ingeniero Junior se estima en 25 €brutos y la tarifa de un Jefe de Proyecto (en este caso el Tutor) es de 40 €/hora, el coste de personal se sitúa en 34.340 €.

En la Tabla A.2 se recogen los costes de material desglosados en equipo informático, herramientas de desarrollo, local de trabajo, documentación y gastos varios no atribuibles (material fungible, llamadas telefónicas, desplazamientos, etc.). Ascienden, pues, a un total de 53.779,65 €.

A partir de estos datos, y teniendo en cuenta que las herramientas de desarrollo así como las

Tabla A.1: *Fases del Proyecto*

<i>Fase 1</i>	<i>Estudio y documentación</i>	<i>350 horas</i>
<i>Fase 2</i>	<i>Implementación del diseño de prueba</i>	<i>120 horas</i>
<i>Fase 3</i>	<i>Desarrollo del demodulador</i>	<i>464 horas</i>
<i>Fase 4</i>	<i>Redacción de la memoria del proyecto</i>	<i>250 horas</i>

Tabla A.2: *Costes de material*

<i>Ordenador de gama media</i>	1.100 €
<i>Dispositivo SFF SDR y licencias</i>	8.372,20 €
<i>Osciloscopio Infinium DSO90604A</i>	41.967,45 €
<i>Local (durante 12 meses, con un coste de 120 €/mes)</i>	1.440 €
<i>Documentación</i>	200 €
<i>Gastos varios</i>	700 €

Tabla A.3: *Presupuesto Final*

Concepto	Importe
Costes personal	34.340 €
Costes material	3.440 €
Costes indirectos (20 %)	6.868 €
Base imponible	44.648 €
I.V.A. (18 %)	8.037€
TOTAL	52.685 €

licencias de los programas utilizados ya han sido atribuibles a los costes de la primera fase de este proyecto, el presupuesto total es el mostrado en la Tabla [A.3](#).

Diseño de Módulo Hardware sobre FPGA

En este apartado se presenta el proceso de diseño de un módulo hardware sobre una FPGA, el planteamiento parte del desarrollo general para luego profundizar en las características propias del entorno de los dispositivos de Xilinx. Con esto se pretende conocer el tipo de elementos que intervienen y las funciones que cumplen en cada una de las fases de diseño, aprender el mecanismo de creación de módulos en VHDL y como estos interactúan con los módulos prediseñados (IP Cores), explicar el proceso de síntesis y la relación que hay entre las diferentes librerías y la simulación del diseño final.

El proceso general de un diseño hardware sobre un dispositivo de puertas programables se divide en tres fases:

- Creación del modelo
- Síntesis
- Implementación

En la primera fase se crea un modelo del diseño que se quiere implementar, para ello se utiliza un editor de texto donde se describe en lenguaje de alto nivel VHDL las características del mismo. Dicha descripción puede contener código que represente circuitos genéricos e independientes de la tecnología del dispositivo a utilizar, instancias de componentes propios de la arquitectura sobre la que se realizará la implementación y también, instancias de componentes que hacen uso de recursos genéricos (CLBs) y que han sido optimizados por el fabricante, estos módulos reciben el nombre de *IP Cores*.

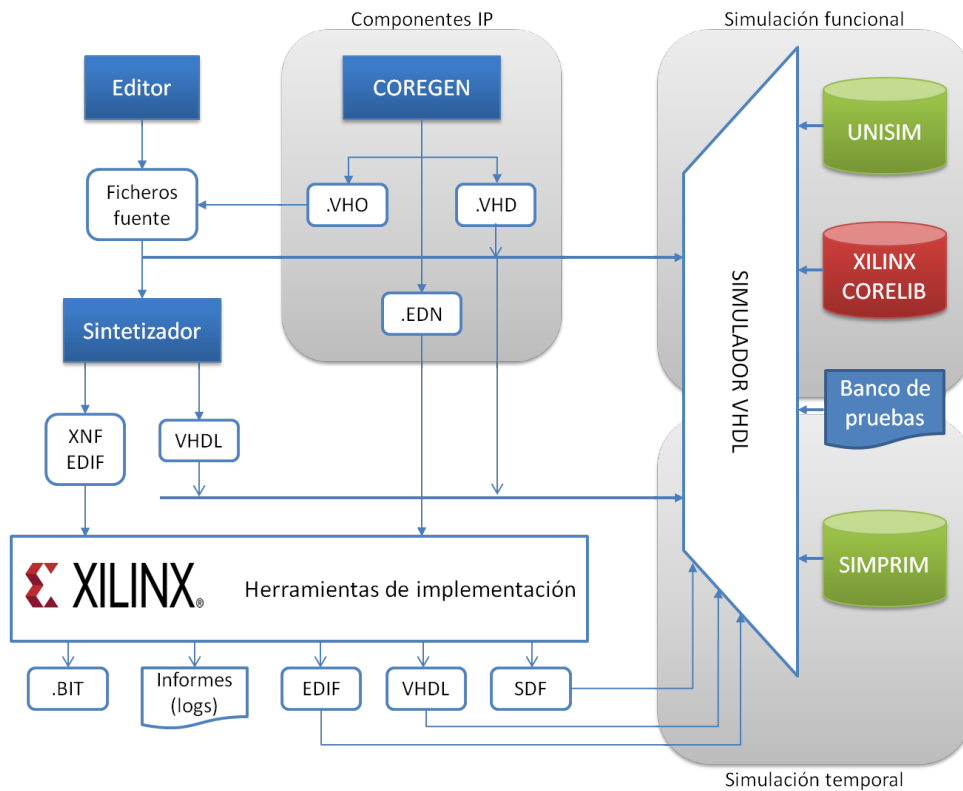


Figura B.1: Flujo de diseño para FPGAs de Xilinx en VHDL.

En la segunda fase se adapta el diseño anterior a un hardware específico teniendo en cuenta la estructura interna del dispositivo, se fijan las restricciones en caso de haberlas y se optimizan las expresiones lógicas con el fin de optimizar los recursos.

En la última fase se programa el diseño en el dispositivo final y se comprueba su funcionamiento.

En la figura B.1 se representa el flujo de diseño sobre un dispositivo de la marca Xilinx. A continuación se detalla este proceso realizado con la herramienta *Xilinx ISE Foundation 9.2i*.

B.1. Creación del diseño

Se comienza con la generación de los ficheros fuente, estos ficheros con extensión `.vhd` contienen la descripción estructural y comportamental de todos los circuitos que integral el diseño así como instancias de módulos de propiedad intelectual que se obtiene a través del COREGEN proporcionado por el fabricante. Una vez se haya comprobado que no hay errores de compilación,

se procede a comprobar su funcionamiento mediante simulación.

B.1.1. Componentes IP

Xilinx proporciona de forma gratuita módulos de propiedad intelectual para sus dispositivos a través del sistema generador de cores que viene incluido en el software de ISE, estos módulos que normalmente describen funciones o circuitos estándar pueden ser integrados en cualquier proyecto o diseño VHDL.

Cuando se crea un componente IP se generan una serie de ficheros de los que se destacan tres, un fichero con extensión *.vhd* que contiene una definición abstracta del módulo y permite la verificación del diseño completo mediante simulación, un fichero con extensión *.vho* que contiene una plantilla con las construcciones necesarias para la declaración e instanciación del componente en el fichero VHDL donde se vaya a utilizar y por último, un fichero con extensión *.edn* que contiene una lista de conexiones que se empleará en la fase de síntesis.

Antes de continuar es necesario verificar el diseño global mediante simulación funcional.

B.1.2. Simulación funcional

Este tipo de simulación se puede realizar con la herramienta ISE ó utilizar cualquier otro programa que permita una simulación avanzada, por ejemplo *Modelsim*. Para llevar a cabo esta tarea es necesario generar un banco de pruebas que valide el diseño. Además, se necesitan modelos VHDL con información perteneciente a los elementos de la librería de la tecnología del dispositivo destino.

Xilinx proporciona varias librerías **VITAL** para realizar simulaciones de diseños para FPGAs. La librería UNISIM es una de ellas, en la que se han desactivado las comprobaciones de violaciones de temporización para realizar simulaciones funcionales con retardos unitarios.

Si el diseño contiene *IP Cores* que utilizan componentes propios de la tecnología destino, será necesario utilizar la librería de simulación *XilinxCoreLib*. Es muy importante tener en cuenta que se ha de agregar el fichero *.vhd* generado por el COREGEN al proyecto y que cualquier modificación sobre el módulo generado obliga a recompilar esta librería.

Existen varias formas de proceder para compilar las librerías y decirle al simulador donde se encuentran, en función de la herramienta que se este utilizando para la simulación, como se describe en [2]. La forma más sencilla de hacerlo es a través del *Simulator Library Compilation*

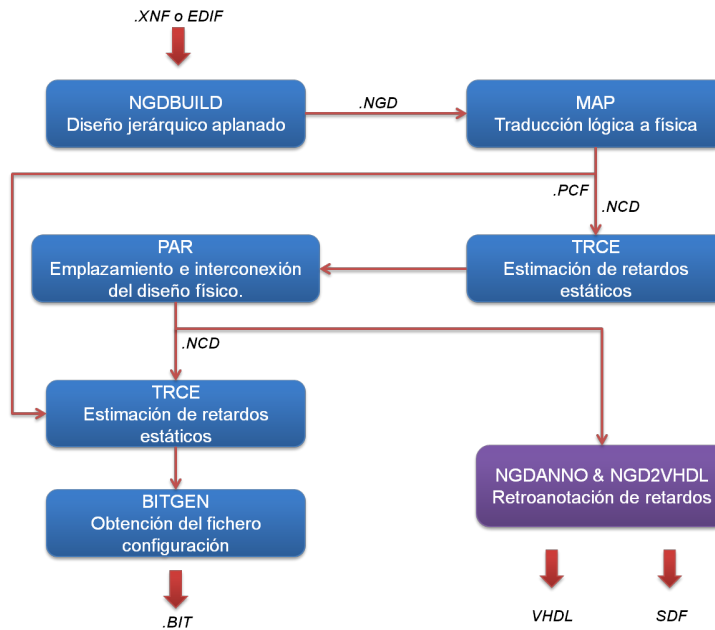


Figura B.2: *Flujo del proceso de implementación.*

Wizard que viene incluido con el software de desarrollo de Xilinx.

B.2. Síntesis

Una vez validado el diseño con la simulación funcional se procede a su síntesis. Esta fase arroja como resultado una lista de conexiones en formato **EDIF** o **XNF**. Si durante el proceso se fijan restricciones topológicas o temporales en el diseño estas se reflejan en un fichero **NCF**.

En caso de querer comprobar los resultados de la síntesis es preciso realizar una simulación funcional con retardos unitarios a nivel de puertas, este tipo de simulación recibe el nombre de simulación post-síntesis. El procedimiento a seguir es similar al anterior y cabe destacar que los componentes IP siguen siendo modelos de comportamiento.

B.3. Implementación

La fase siguiente es la de implementación del diseño utilizando las herramientas que Xilinx proporciona para ello. En la figura B.2 se muestra el diagrama de flujo de las fases en las que se divide la implementación del diseño y las herramientas utilizadas en cada una de ellas.

En este punto se tiene como entrada el fichero que generó la fase de síntesis (XNF o EDIF),

dicho fichero es tratado por la herramienta NGDBUILD que se encarga de realizar la fase de traducción (*translate*), aquí se enlazan los ficheros de lista de conexiones para crear un único fichero de salida en el que queda reflejada toda la jerarquía del diseño. El formato de este fichero se denomina NGD (*Native Generic Design*) y contiene componentes lógicos tales como biestables, puertas lógicas, RAMS, etc.

En la fase de asignación (*Map*) los componentes lógicos se asignan por medio de la herramienta MAP a los componentes físicos de la FPGA (LUTs, biestables, triestados, etc). Posteriormente se encapsulan dichos componentes en los bloques configurables (CLBs) del dispositivo y se genera un fichero de salida en formato **NCD**. La herramienta TRCE realiza un análisis estático de tiempos en el que se estudian los caminos críticos de las señales comprobando el cumplimiento de las restricciones temporales introducidas en el fichero PCF (Project Constraints File).

En la fase llamada *Place & Route* la herramienta PAR realiza el emplazamiento y la interconexión de los bloques configurables en la FPGA y genera una lista de conexiones en formato NCD. Una vez finalizado este proceso se dispone de toda la información sobre los retardos internos.

La fase de sincronización cuenta con dos herramientas para obtener toda la información sobre simulaciones temporales del diseño. La primera de ellas (NGDANNO) realiza la retroanotación con la información de retardos internos, la segunda (NGD2VHDL) crea un fichero VHDL estructural en base a los componentes físicos de la FPGA y un fichero SDF con los retardos internos de esta.

La fase final es la fase de configuración que a través de la herramienta BITGEN genera el fichero con extensión *.BIT* que será cargado en la FPGA.

Bibliografía

- [1] BERNAL, DAVID; CLOSAS, PAU Y FERNANDEZ-RUBIO, JUAN. Digital I&Q Demodulation in array processing: Theory and Implementation. In *European Signal Processing Conference*. (Agosto 2008).
- [2] FPGARELATED.COM. Página web. <http://www.fpgarelated.com/usenet/fpga/show/74367-1.php>.
- [3] HO, K. C, ET AL. A digital quadrature demodulation system. *IEEE Transactions on, Aerospace and Electronic Systems [en línea]*. Vol. 32, no. 4 (1996), pp. 1218–1227.
- [4] JIMENEZ, VÍCTOR P, ET AL. Design and implementation of synchronization and AGC for OFDM-based WLAN receivers. *IEEE Transactions on Consumer Electronics [en línea]*. Vol. 50, no. 4 (2004), pp. 1016–1025.
- [5] LUHMANN, NIKLAS. What is Communication?. Communication Theory. *Wiley Online Library*. Vol. 2, no. 3 (1992), pp. 251–259.
- [6] LYRTECH INC. Lyrtech Development Platform Command Shell User’s Guide, version 1.6, Octubre 2007.
- [7] LYRTECH INC. SFF SDR API guide version 1.0, Enero 2007.
- [8] LYRTECH INC. SFF SDR Development Platform User’s guide version 1.1, Octubre 2007.
- [9] REAL ACADEMIA ESPAÑOLA. (S. F.). Página web. <http://www.RAE.es/>.

- [10] RICE, D. W; WU, K. H. Quadrature Sampling with High Dynamic Range. *IEEE Transactions on, Aerospace and Electronic Systems [en línea]*. Vol. AES-18, no. 6 (1982), pp. 736–739.
- [11] ROOME, S.J. Analysis of quadrature detectors using complex envelope notation. *IEEE Proceedings F, Radar and Signal Processing [en línea]*. Vol. 136, no. 2 (1989), pp. 95–100.
- [12] SAMUELI, HENRY Y WONG, BENNETT. A VLSI architecture for a high-speed all-digital quadrature modulator and demodulator for digital radio applications. *IEEE Journal on, Selected Areas in Communications Vol. 8*, no. 8 (Octubre 1990), pp 1512–1519.
- [13] TEXAS INSTRUMENTS. 16-BIT, 500 MSPS 2x-8x Interpolating Dual-Channel Digital-To-Analog Converter (DAC), Septiembre 2006. Disponible en web: <http://www.ti.com/lit/ds/symlink/dac5687.pdf>.
- [14] TEXAS INSTRUMENTS. Code Composer Studio Development Tools v3.3. Getting Started Guide, Octubre 2006. Disponible en web: <http://www.ti.com/lit/ug/spru509h/spru509h.pdf>.
- [15] VALVERDE, CARLOS. IMPLEMENTACIÓN DE UN SISTEMA OFDM EN UN DISPOSITIVO SFF SDR. Tech. rep., Universidad Carlos III de Madrid, Departamento de Teoría de la Señal y Comunicaciones, 2010.
- [16] XILINX. ISE Quick Start Tutorial, 2007. Disponible en web: http://www.xilinx.com/support/sw_manuals/xilinx92/download/.
- [17] XILINX, LOGICORE. *DDS v5.0, Product Specification*, Abril 2005.
- [18] XILINX, LOGICORE. *Distributed Arithmetic FIR Filter v9.0, Product Specification*, Abril 2005.
- [19] XILINX, LOGICORE. *Multiplier v10.0, Product Specification*, Abril 2007.